THE USENIX ASSOCIATION NEWSLETTER

;login:

DO NOT REMOVE
FROM COAST LAB

OCTOBER 1994 • VOLUME 19 • NUMBER 5

CONTENTS

FROM THE EDITOR

Woodstock II 3
Letters to *;login:* 3

USENIX NEWS

President's Letter 4
High Speed Networking Symposium Report *by Casey Leedom* 8
Summer '94 Conference Reports *by Peter Collinson, Jerry Peek, Lou Katz* 16
USENIX C++ Conference Report *by Susan E. Waggoner Shopiro and Jonathan Shopiro* 23
USENIX Supports Student Participation 25
Community News 25

SAGE NEWS

On Professionalism *by Tom Limoncelli* 26
Surviving Solaris 2.X *by Hal Pomeranz* 27
Your Voice *by Win Bent* 28
System Administration Models *by Elizabeth Zwicky* 29

FEATURES

Understanding UNIX Workstation Performance *by Scott Hazen Mueller* 32
What's New 35

STANDARDS REPORTS

An Update on Standards Relevant to USENIX Members *by Nick Stoughton* 43

BOOK REVIEWS

The Bookworm *by Peter H. Salus* 47

ANNOUNCEMENTS & CALLS

Symposium on Very High Level Languages 49
Symposium on Operating Systems Design and Implementation (OSDI) 51
Calls for Papers:
- SANS IV 53
- Symposium on Mobile and Location-Independent Computing – 1995 55
- UNIX Security 57
- COOTS 59
- LISA IX - 1995 61

ETC...

Book Publishers Discount Order Forms 63
Local User Groups 72
Calendar of Events 74

USENIX The UNIX and Advanced Computing Systems Professional and Technical Association

;login: is the official newsletter of the USENIX Association.

•*;login:* (ISSN 1044-6397) Volume 19, Number 5 (October 1994) is published bimonthly by the USENIX Association, 2560 Ninth Street, Suite 215, Berkeley, CA 94710.

• $24 of each member's annual dues is for an annual subscription to *;login:*. Subscriptions for nonmembers are $50 per year.

• Second-class postage paid at Berkeley, CA and additional offices.

• POSTMASTER: Send address changes to *;login:*, USENIX Association, 2560 Ninth Street, Suite 215, Berkeley, CA 94710.

EDITORIAL STAFF

• Rob Kolstad, Editor <*kolstad@usenix.org*>

• Bryan McDonald, SAGE Editor <*bigmac@erg.sri.com*>

• Nick Stoughton, Standards Report Editor <*nick@usenix.org*>

• Carolyn S. Carr, Managing Editor & Typesetter <*carolyn@usenix.org*>

• Monica Jensen, Copy Editor

• Lilia Scott, Proofreader

MEMBERSHIP AND PUBLICATIONS

USENIX Association
2560 Ninth Street, Suite 215
Berkeley, CA 94710
Telephone: 510/528-8649
FAX: 510/548-5738
Email: <*office@usenix.org*>

CONTRIBUTIONS SOLICITED

You are encouraged to contribute articles, book reviews, and announcements to *;login:*. Send them via email to <*login@usenix.org*> or through the postal system to the Association office. Send SAGE material to <*bigmac@erg.sri.com*>. The Association reserves the right to edit submitted material. Any reproduction of this newsletter in its entirety or in part requires the permission of the Association and the author(s).

© 1994 USENIX Association. USENIX is a registered trademark of the USENIX Association. UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Limited. The Association acknowledges all other trade references made herein.

;login: is produced with Framemaker 4.0 software provided by Frame Technology and displayed on an NCD X Terminal, donated by Network Computing Devices. The system is served by a Sun SPARCsystem 10 server. Output is sent to a 600 dpi QMS 860 laser printer, donated by Quality Micro Systems. The newsletter is printed on recycled paper.

SAVE THESE DATES

NOVEMBER 14–17, 1994

USENIX SYMPOSIUM ON OPERATING SYSTEMS DESIGN AND IMPLEMENTATION (OSDI)

Co-sponsored by ACM SIGOPS and IEEE TCOS

acm. IEEE COMPUTER SOCIETY

Monterey, California

Program Chair: Jay Lepreau, *University of Utah*

Papers Due: October 4, 1994

APRIL 10–11, 1995

2ND USENIX SYMPOSIUM ON MOBILE & LOCATION-INDEPENDENT COMPUTING

Ann Arbor, Michigan

Program Chair: Jim Rees, *CITI, University of Michigan*

Abstracts Due: January 2, 1995
Authors Notified: January 23, 1995
Papers Due: March 6, 1995

JANUARY 16–20, 1995 – REGISTER BY DEC. 9 AND SAVE $100!

1995 USENIX TECHNICAL CONFERENCE ON ADVANCED COMPUTING SYSTEMS

New Orleans, Louisiana

Program Chair: Peter Honeyman, *CITI, University of Michigan*

Papers Due: November 14, 1994

Keynote Address: **Ubiquitous Computing**
by Dr. Mark Weiser, *Xerox Palo Alto Research Center*

LEARN TOOLS, TACTICS AND TECHNIQUES THAT WILL ENHANCE YOUR KNOWLEDGE AND IMPROVE YOUR SKILLS.

Hear never-before-published work.

Improve your skills and advance your career.

Meet your peers and discuss common problems and issues.

Get input from industry experts in small informal sessions.

Learn tools and techniques that allow you to earn.

SIGN UP NOW. CHOOSE FROM 20 TUTORIALS, INCLUDING 8 NEVER-BEFORE-OFFERED!

MEET THE VENDORS AND TALK TURKEY FREE ENTRY TO VENDOR DISPLAY

JUNE 5–7, 1995

5TH USENIX UNIX SECURITY SYMPOSIUM

Sponsored by the USENIX Association, in cooperation with: The Computer Emergency Response Team (CERT), IFIP WG 11.4 (pending), and UniForum (pending).

Salt Lake City, Utah

Abstracts: February 13, 1995
Authors Notified: March 8, 1995
Papers Due: May 1, 1995

JUNE 26–29, 1995

USENIX CONFERENCE ON OBJECT ORIENTED TECHNOLOGIES (COOTS)

Monterey, California

Program Chair: Vince Russo, *Purdue University*

Tutorial Program Chair: Doug Lea, *SUNY Oswego*

Abstracts Due: March 6, 1995
Authors Notified: April 3, 1995
Papers Due: May 15, 1995

PHONE, FAX OR EMAIL US FOR MORE INFORMATION

Please contact: USENIX Conference Office
22672 Lambert St., Suite 613, Lake Forest, CA USA 92630
Phone: +1 714 588 8649; FAX: +1 714 588 9706; Email: conference@usenix.org

For more information about these events, access the USENIX Resource Center on the World Wide Web. The URL is **http://www.usenix.org**. Or send email to our mailserver at: **info@usenix.org**. Your message should contain the line: send catalog. A catalog will be returned to you.

FROM THE EDITOR

Woodstock II

The 25th anniversary of Woodstock has come and gone along with its progeny, the pair of Woodstock II's.

I was in high school for Woodstock I and was attending an NSF summer institute for high school computer jockeys when the Woodstock album was released (and a wonderful album it was). Life was full of possibilities, dreams, and unlimited potential back in 1970. That summer, we talked of peace and war, religion and church, relationships, and technology. As it turns out, 70% of the 102 participants were Jewish, 29% Catholic, and 1% "undecided". We had marvelous discussions well into the night, learning of each other's beliefs and customs.

I'm not in high school any more, but sometimes I yearn to go back to the kind of interaction we had that summer. Why? Because it was not divisive and fraught with the context of "winners" and "losers" as I perceive today's interactions so often are. We were just random kids trying to figure things out.

Nowadays, it seems, issues of all kinds are dissected for us by pundits, analysts, and citizens. There is a "right way" (invariable the author's way) and – here's the catch – all other ways are "the wrong way". In my community, the bumper stickers that say, "Don't blame me, I voted for Bush" appeared before our current President even took office! Not exactly what I would call a "wait and see" attitude.

I write this column to inspire people to opt for constructive criticism, real problem solving, and a certain kind of attitude about compromise and tolerance that is different, say, than the attitudes of the various baseball partisans as reported in early August. We can all be winners – there need not be losers.

We must all learn to get along. My state is being splintered and faction-ized so much that no thought is being given to anything but trivial issues. This can't be healthy.

As a society, we have to do better. I hope our technical community, as a part of the larger society, can be part of the solution.

- RK

Letters to *;login:*

Re: Conquering Corporate Computing with Message-Oriented Middleware

by Michael A. Palombo
<mpalombo@netcom.com>

Dear *;login:*,

I found Tim Daneliuk's feature "Conquering Corporate Computing with Message-Oriented Middleware" in the August issue of this newsletter rather misleading. I don't feel Mr. Daneliuk's message reflects dishonesty but rather a narrow knowledge of the issues surrounding distributed processing (DP) and distributed transaction processing (DTP).

continued on page 5

USENIX ASSOCIATION BOARD OF DIRECTORS

If a member wishes to communicate directly with the USENIX Board of Directors, he or she may do so by writing to *<board@usenix.org>*.

President:
• Stephen C. Johnson *<scj@usenix.org>*

Vice President:
• Eric Allman <eric@usenix.org>

Secretary:
• Lori Grob <grob@usenix.org>

Treasurer:
• Rick Adams *<rick@usenix.org>*

Directors:
• Tom Christiansen *<tchrist@usenix.org>*
• Dan Geer *<geer@usenix.org>*
• Andrew Hume *<andrew@usenix.org>*
• Greg Rose *<ggr@usenix.org>*

Executive Director
• Ellie Young *<ellie@usenix.org>*

CONFERENCE & SYMPOSIA

• Judith F. DesHarnais
Conference Coordinator
USENIX Conference Offic
22672 Lambert Street, Suite 613
Lake Forest, CA 92630
Telephone: 714/588-8649
FAX: 714/588-9706
Email: *<conference@usenix.org>*

• Peter Mui
Vendor Displays
Email: *<display@usenix.org>*

TUTORIALS

• Daniel V. Klein
Tutorial Coordinator
Telephone: 412/421-2332
Email: *<dvk@usenix.org>*

WWW URL: *http://usenix.org*

AUTOMATIC INFORMATION SERVER

To receive information electronically about upcoming USENIX symposia & conferences finger *<info@usenix.org>* and you will be directed to the catalog which outlines all available information about USENIX services.

SUPPORTING MEMBERS

ANDATACO
ASANTÉ Technologies, Inc.
Frame Technology Corporation
Matsushita Electrical Industrial Co., Ltd.
OTA Limited Partnership
Quality Micro Systems, Inc.
SunSoft Network Products, Sun Microsystems, Inc.
Tandem Computers, Inc.
UUNET Technologies, Inc.

The closing dates for the next two issues of *;login:* are October 12, and December 14,1994.

President's Letter

by Steve Johnson
<scj@usenix.org>

USENIX is not just operating systems! As many of you know, we have had a series of C++ workshops and conferences that drew the premier implementors and, I feel, played a significant role in bringing C++ to the stage of standardization. We are broadening this conference to address a variety of issues in object-oriented languages, including C++, successors to C++ and others. We have also played a significant role in the spread of perl and Tcl throughout the technical community with our tutorials, papers, invited talks, and BOF sessions.

In October, we will be sponsoring another workshop in the languages area, on Very High Level Languages (VHLL). [*See page 49 for the preliminary program*]. The intent is to focus on what it is that languages such as perl, Tcl, REXX, ML, etc. have in common, what problems they are trying to solve, how well they solve them, and what tools and techniques have been especially useful. Our call for papers drew over fifty responses, and we have put together an interesting workshop – being in Santa Fe, NM is certain to make the experience even more pleasant.

Underlying our interest in this topic is a real concern about software productivity. As an adolescent, programming my first computer using punched cards, I could compile and execute a program, look at the output, make some changes, and resubmit in a cycle that was typically about 1/2 to 1 hour. This compile/test/edit cycle seems to me to be one of the key measures of software productivity.

Using a modern workstation, I can go through a similar cycle in a minute or two. Being charitable, I call this two orders of magnitude – improvement in productivity. If I account for the fact that the size of the programs being dealt with is much larger than in 1970, I might give myself one or two more orders of magnitude. Being very charitable, I see the software process at most, four orders of magnitude better than it was in 1965 (and I think most old-timers would say that two or three orders of magnitude feels more realistic).

Over the same period, the cost of machine cycles has dropped by roughly seven or eight orders of magnitude. More seriously, the actual cost of computers has dropped from millions of dollars to hundreds or thousands of dollars. Users are not willing to spend several times the hardware cost of their system in order to populate it with software.

The result is a crunch in the software field. The cost of putting a UNIX release in the field can easily run into the tens (or even hundreds) of millions of dollars. The traditional software methods (huge programmer teams, large investment in source code control, bug tracking and testing) will simply not be affordable by most companies in a very few years.

One approach is to stretch out these techniques by lowering the labor cost – there are a number of countries (India and Russia have been in the news recently) that have a lot of skilled programmers willing to work for a fraction of US wages. By doing business as usual, some companies hope to stretch out the traditional techniques for a few more years.

But the real solution to this problem lies in using the same cheap cycles that are causing the problem, to help solve it. Use of interpreters, programming environments, and visual shells give improvements in productivity the same way that moving from assembler to C, or using *make*, gave improvements in productivity at the cost of using more machine cycles. As before, there will be many people (and companies) who'll say they "can't afford" these techniques. And as before, they will be wrong. I don't think the October VHLL conference will be the last one. . .

USENIX MEMBER BENEFITS

As a member of the USENIX Association, you receive the following benefits:

- Free subscription to *;login:*, technical features, system administration tips and techniques, international calendar of events, SAGE News, media reviews, Snitch Reports from the USENIX representative and others on various ANSI, IEEE, and ISO standards efforts, and much more.
- Free subscription to *Computing Systems*, the refereed technical quarterly published with The MIT Press.
- Discounts on registration fees for the large, multi-topic Winter technical conference, the System Administration conference (LISA), and the various single-topic symposia addressing topics such as object technology, security, operating systems, high-speed networking, and mobile computing – as many as eight technical meetings every year.
- Special subscription rates to the periodicals *Open Computing* (the new name for *UNIX World*) and *The LINUX Journal*.
- Discounts on proceedings from USENIX conferences and symposia and other technical publications. Discounts on the USENIX Association book published by The MIT Press. Now available: *The Evolution of C++: Language Design in the Marketplace of Ideas*, edited by Jim Waldo of Sun Microsystems Laboratories.
- Discounts on the five volume set of 4.4 BSD manuals plus CD published by O'Reilly & Associates and USENIX.
- Savings (10-20%) on selected publications from McGraw-Hill, The MIT Press, Prentice Hall, John Wiley & Sons, O'Reilly and Associates, and UniForum.
- Right to join SAGE, the System Administrators Guild

To become a member or receive information regarding your membership status or benefits, please contact *<office@usenix.org>* or telephone (510)528-8649

(Letters to ;login: continued from page 3)

By illustrating an argument containing some basic correct assumptions he creates a false impression of overall expertise in distributed technology. The broad issue of middleware as a unifying force goes beyond the issues of RPC vs. message-oriented front-end APIs. There are a variety of issues he does not consider including OS architecture of the key distributed service providers, interoperability of component technology and standards supported by the middleware, the role of transaction processing monitors, and the role of distributed service providers.

In addition, his discussion of interfaces should have contained a balanced analysis of message-oriented client-server, remote procedure calls (RPCs) and point-to-point interfaces (for example, XATMI vs. TxRPC vs. CPI-C) from an application programmer and technical architecture perspective. The actual discussion of message-oriented middleware interfaces versus RPCs is largely incorrect (or misleading). This analysis forms the basis for the author's erroneous assertion that "RPCs have failed to conquer the large enterprise". In fact, RPCs are now doing very well in the enterprise and this is reflected in the market share growth of DTP systems like Encina and CICS/6000 and the rapid growth of DCE after only a year of production availability.

Unlike ONC, DCE is designed for corporate computing and includes key services needed to support and operate a secure distributed environment. DCE is more than RPCs. Sun is supporting DCE but cannot abandon ONC which is used in NFS. Regarding DTP systems, the Gartner Group predicts Encina and CICS/6000 will account for the majority of the DTP market in 1997.

As OS technology evolves, this trend toward RPC should continue as more products take advantage of multitasking and multithreading. I agree with Mr. Daneliuk that connectivity with legacy systems is critical. IBM is offering DCE with its MVS product line and will offer DCE applications servers to distribute CICS and IMS functionality. Using CPI-C, X/Open's peer to peer standard, the Encina DTP system can coordinate two phase commit protocol with LU6.2 enabled applications including CICS, IMS, and DB2. CICS/6000 obviously offers superior downsizing opportunities for mainframe CICS applications.

The UNIX community has done great things to promote DP and DTP and this is reflected in the overall market. The current DTP market share leader, Tuxedo (a message based system), came out of AT&T's UNIX community; its main competitors, RPC based DTP systems, also come from the UNIX community. IBM has adopted DCE technology for OS/2, MVS, VM and the OS400 and Microsoft MS RPC is based on the DCE RPC and distributed service model (with non-secure interoperability).

Response by the Author

by Tim Daneliuk
<tundra@ct.covia.com>

Mr. Palombo's comments reflect many oft heard notions about distributed computing and bear a fair-minded response. I'll speak to them in the order originally stated.

I'd like to begin by addressing the question of my "narrow knowledge" in the field. My employer, Covia Technologies, is a subsidiary of Galileo International. Galileo owns and operates the world's largest airline reservation (transaction processing) system. This system includes the entire United Airlines Apollo travel system as well as providing similar services for Air Canada, British Airways, KLM, SwissAir, USAir, Al Italia, and a host of other travel providers. I have been both an implementor and corporate architect in this environment.

Our system complexity and service-level requirements far exceed anything typically seen in the majority of distributed systems. For example, we can provide peak throughputs on the order of 20,000 TPCA equivalent transactions per second. Keep in mind, that this means virtually 100% availability with sub-3 second response times to our 100,000+ users around the world, 24 hours a day, 365 days a year. Yes, Mr. Palombo, I do understand something about the complexities of large, distributed TP environments. As we'll see in a moment, the technologies argued for are simply incapable of doing things on this scale.

As a first step, we need to debunk the notion of RPC vs. Message Oriented Middleware as being an API issue. API convergence is not the hard part. The heart of the discussion is that of service semantics. RPCs are a poor choice for building large distributed systems with centralized data store and control precisely because of their inherent synchronous semantics. Passing control to a remote server and then spin-locking while you wait for an answer is slow, possibly unreliable, hard to recover, and very connection intensive. This has *nothing* to do with APIs and everything to do with the underlying service paradigm.

I'll grant Mr. Palombo that I failed to cover the alphabet soup of interfaces presently being promoted. There are several reasons for this, not the least of which was time and space constraints in the original article. However, I feel compelled to briefly speak to the question "open" interfaces. Until several months ago, I was my company's representative to the X/OPEN XTP working group. I have seen hundreds of man hours expended in the discussion of "XATMI vs. TxRPC vs. CPI-C" and yet we still don't really have a standard. In fact, we have the opposite, we have *every* standard. Why? Those who make their living creating "open" standards would claim that different interfaces are needed to serve different problem classes. Fair enough. However, what is rarely said is that standards committees

are staffed by people from real companies all of whom are seeking competitive advantage. It is unrealistic to believe that the major players in industry are really seeking a uniform set of interfaces. (If you don't believe this look at the variations on the UNIX theme that exist, particularly above the base C library and X-Windows.)

In fact, no matter what their marketing literature says, the industry leaders (yes, even the UNIX vendors) are interested in paying lip service to the standards and then "value adding" their systems in a non-standard way to lock their customers up forever. The reality here is that standards which prevail are those defined by users not by architectural committees.

I take issue with the statement that ". . . RPCs are doing very well in the enterprise . . ." RPCs are doing well in a very limited arena: the arena of the small. If all you need to do is divide and conquer applications for a few hundred users over several LANs, RPC may well be the right model. This is particularly true where the reliability and service level requirements are not too stringent. This tends to be the case for departmental computing, for example. The *enterprise*, by contrast, is characterized by tens-of-thousands of users who usually need access to some logically central data store which the corporation also wants controlled centrally. Try nailing up 25,000 TCP/IP connections to a mainframe to service RPCs calls from all over your distributed enterprise and watch what happens to your system. In fact, this is *not* what is going on in Corporate America. Real-world, high-volume TP shops are moving to messaging, distributed queues, and generally asynchronous computing models.

The great reality of corporate computing is that: a) There will always be a legacy and b) There will always be heterogeneity in networks, operating systems, and protocols. It is sheer fantasy to believe that the multi-billion dollar investment in place today will be ripped out so that everyone can converge to TCP/IP, UNIX, CPI-C, DCE, and so on. There are business applications in place today (payroll springs to mind) that have been running in some form for 30 years and are not going to get rewritten just because someone got a dose of the latest computing theology.

I share Mr. Palombo's view that the UNIX community has done much to promote distributed computing. UNIX is my personal environment of choice both professionally and at home. Nonetheless, I am deeply disappointed that the UNIX vendor and research community is *not* connecting with the needs of the real-world. The all-or-nothing mentality which pervades the UNIX community has minimized its usefulness in the large corporate setting. The lack of attention to reliability, predictable service levels, and very large scaling is leaving the door open for vendors of proprietary solutions and *THEY* are winning the commercial battle.

In Response to "Optimizing Your Shell Scripts"

by Chet Ramey and Elizabeth A. Ruzga
<chet@odin.ins.cwru.edu>

Hi Scott,

I have a few comments about your article "Optimizing Your Shell Scripts" in the last (August) issue of *;login:*. The article is pretty good, and full of useful advice, but I think there are several inaccuracies.

1. In the "Fork/Exec Considered Harmful" section, the script using `set` does not do the same thing as the fragment using `awk`. The difference is in the way that `sh` and `awk` handle multiple consecutive field delimiters. `awk` produces null fields; `sh` never does (it considers multiple consecutive delimiting characters to be a single delimiter). If you ran both of these fragments on a password file with an empty field, they would produce different results. The `awk` version always outputs what is expected; the `sh` version does not.

The passwd file was just an example, but there is a fundamental difference in how `sh` and `awk` split fields that could come back to bite someone.

The POSIX.2 committee has standardized a combination of the `sh` and `awk` behavior for future *sh* versions.

2. There is a misunderstanding in how assignment statements preceding builtin command names affect that command.

On my SunOS and BSD/386 machines, the `sh` version of the password parsing script does not work at all, due to the way the assignment to IFS is handled. The shell saves the assignments to pass to the command via the environment; they do not affect the expansion of the rest of the line. Even `eval` does not produce the right effect. POSIX.2 standardizes this behavior.

With the following input file (named "input"):

```
chet:*:286:10:Chet Ramey:/usr/homes/chet:/bin/
bash
ear8::40420:100:Elizabeth A. Ruzga:/u/20/ear8:/
usr/bin/fn
```

and these scripts:

```
$ cat s1
 while read pwdline; do
  if test ! -z "$pwdline"; then
      IFS=: set $pwdline
            gcos=$5
```

```
    fi

echo "$gcos"
  done
```

```
odin(2)$ cat s2
while read pwdline; do
    gcos=`echo $pwdline | awk -F: '{print $5}'`
          echo $gcos
done
```

I get the following results on SunOS 4.1.2 and BSD/386

```
$ sh ./s1 < input

/u/20/ear8

odin(2)$ sh ./s2 < input
```

3. In sh, "set" is a "special" builtin, meaning that assignments preceding it on the command line remain in force after the command completes. Thus the only portable way to make the fragment using "set" work at all is to save and restore $IFS around the "set" command. That still does not handle the problem with null fields.

This is why executing "s1" above results in something being printed from the second line.

Even changing the above script to save and restore $IFS does not fix the problem with null fields:

```
$ cat s3
while read pwdline; do
  if test ! -z "$pwdline";then
      oifs="$IFS"
      IFS=:
      set $pwdline
      gcos=$5
       IFS="$oifs"
  fi

   echo $gcos
done
$ sh ./s3 < input
Chet Ramey
/u/20/ear8
```

POSIX.2 standardizes this behavior as well.

4. The final loop you present in your "Use Filters as Filters" section wastes two unnecessary processes (two extra forks, one extra exec). A better solution for this particular problem is to discard sed entirely and use "case":

```
while read line; do
    case "$line" in
     \#*|"") ;;
  *)  echo processing "$line";;
 esac
done
```

Not enough scripts use this pattern-matching ability.

5. Every shell that provides a builtin "test" provides a builtin version of the "[" synonym. This includes the SVR2, SVR3, and SVR4 versions of /bin/sh, ksh, bash, and zsh. The BSD shells have never had "test" builtin.

Response to response

by Scott Hazen Mueller
<scott@zorch.sf-bay.org>

Chet – thanks for your comments. I really appreciate the input.

I'd like to mention a few general principles in response:

1. Always test code in your own environment. Some features won't work on every system, and I did my testing for this article on a real oddball.

2. Always test your corner cases. Error handling can slow code down, but the results of not testing your input against your expectations can be disastrous.

3. Know your input data. If you know you don't have any corner cases, or filter effectively for them, you can then cut corners. If you do this, document your assumptions.

USENIX High-Speed Networking Symposium Report

by Casey Leedom
<leedom@sgi.com>

The symposium, held in Berkeley, California, August 1-2, 1994, was well-attended with 220 attendees. There were many fine presentations, and we were also blessed with keynote addresses by Craig Partridge and Van Jacobson. I think everyone came away feeling that they had spent their time in a valuable manner.

This symposium is part of a series of special subject symposia that USENIX sponsors. As such, it isn't an annual event and may not even be repeated. Personally, I hope that it is repeated, perhaps every two years. It was well-organized and presented the work being done from a higher perspective than "the trenches," which is what you often get at InfoCom, SigCom or the other annual networking conferences. I think both have their place. As an added bonus, many of the speeches and slides for the presentations will be available via the World Wide Web (WWW) at the Internet University, *http://town.hall.org/university/index.html.*

That aside, let's get on with the details.

Monday, August 1:

Keynote address: High Performance Systems and Protocols – Myths and Realities

Craig Partridge, Bolt, Beranek and Newman, Inc.

The thesis of Craig's presentation is that there are some widely accepted myths about gigabit networking and these myths are going to get us into trouble as we start assembling "The Information Superhighway." (I promise, that's the last time I will use that term in this report!) These myths are:

1. A gigabit/second is fast,
2. TCP is a heavyweight protocol,
3. UNIX cannot support high-performance, real-time applications, and
4. Voice, video and data will aggregate well. In particular, the myth is that it will all just look like a big telephone network and that we can blindly use the knowledge we've acquired from the current telephone network.

There's some truth that a Gb/s is fast since many current commercial products are struggling to achieve 100Mb/s. However, even moderately fast personal computer CPUs can already chew that much up: 32bits x 33Mhz is more than 1 Gb/s. Observe that the implications of the myth are painful, because they discourage the rapid deployment of gigabit products on the grounds that gibabits are a niche need. There are other bottlenecks as well; some of the more important include memory systems, buses, and peripheral devices. Memory is getting faster far slower than CPUs are getting faster. In fact, it's really not much faster than it was five years ago. We're still putting 70ns memory in our systems! Memory is getting wider (but there's only so far we can push this) and hybrid devices are appearing. Commercial buses are just now hitting 1Gb/s. Peripherals aren't even close, coming in at low 100s of Mb/s.

The message is that we're rapidly getting to the point where further system speed increases are going to be difficult simply because we can't feed the new faster CPUs! And 1 Gb/s on the wire is not fast compared to future system speeds.

The myth that TCP is a heavyweight protocol is an odd and unfortunately long lived one. Odd because there is no formal definition of "lightweight" (it mostly seems to mean "not TCP") and long lived because people don't benchmark carefully to separate out effects of disk drives, buses, virtual memory, etc. There's been a lot of work done to speed up TCP, notably by Van Jacobson. TCP and IP header handling is now down to about 100 instructions and the check sum is basically free when interleaved with data copying.

Data handling is typically the major cost in packet processing. The worst TCP implementations do five, six, and even seven copies! The best implementations do 1 copy using interfaces with large amounts of memory. For sufficiently large packets, TCP should run at memory speed. Moreover, implementations already exist: Cray (Dave Borman), HP Snake with AfterBurner (based on Van Jacobson's WITLESS network interface architecture) and Post 4.4BSD-lite by Van Jacobson. (See *IEEE Network*, pp. 36-43, July 1993 for an excellent article on the AfterBurner. The rest of that issue is devoted to high-speed networking so it's well worth looking up.)

The contention that UNIX cannot support high-performance, real-time applications boils down to the following debate: What kind of devices do we hook up, general purpose computers or special purpose, service specific boxes ("set-top boxes")?

One real-time model has the net dumping incoming packets into a buffer that the application picks up, processes, and puts on an output device exactly when the output device is supposed to play or display the contents of the packet. This model is very hard on UNIX (and other general purpose operating systems). It places hard requirements on the scheduler and cries out for an embedded real-time system. The implication is: use a set-top box instead of UNIX.

A different real-time model has the net dumping incoming packets into a buffer, the application picks them up, processes them and dumps them into a second buffer and the

output device picks up the processed packets as it needs them. Process scheduling thus becomes far simpler, requiring only a few more smarts from the output device. Why should the application wake up every 1/30 second? Insisting on that model dictates the architecture of the solution!

The final myth that voice, video, and data will aggregate well is a different kind of myth. The first three were engineering issues; this is a modeling problem. The main source of this problem is researcher weakness. Researchers want to treat everything as Poisson distributed; the math is easy and we know how to solve the problem. Within a short time, Poisson distributed traffic will average out into a virtually constant rate. Unfortunately data traffic doesn't look like this. It looks fractal. Even at very large time scales you still see big surges and lulls in traffic rate.

The implication is that data requires a lot of buffering in the network. The telephone model will not suffice for a single general network, and the telephone companies that are pursuing such a solution are in for some serious surprises!

In Craig's concluding remarks he summarized his talk as follows: when will high-speed networking become affordable and available? Probably quite soon, at least in the local case. What protocols will we use? TCP isn't a bad choice. What kinds of devices will we plug into the network? UNIX and other multiprogrammed operating systems should work fine. Who gets to determine the future? No one person or group really knows enough answers to be a clear leader.

In the questions afterwards, Mike O'Dell asked Craig how we are going to deal with delay and jitter if we make the network buffers really large, as Craig suggests. Craig responded that bigger buffers on receivers would allow them to smooth jitter, and smarter queuing in the network could control delay. Personally, I remained unconvinced about our ability to control delay (smoothing jitter is easy but adds even more delay). As data networking researchers, I think we may be making the same hubristic mistake Craig accuses the Telephony people of making, namely that everything can be made to operate under a data networking model. Again, no one knows enough to answer yet. Besides, I think we sometimes forget that while the idea of a single unified voice, video, and data network may be a desirable goal, the ability to achieve it is still in question.

Protocol Architectures

Session chair: Jeffrey Mogul, Digital Equipment Corporation

Modular Communication Subsystem Implementation Using a Synchronous Approach

Claud Castelluccia, INRIA

Claud talked about his and Walid Dabbous' work on creating application specific protocol stacks. This approach seeks the flexibility to create protocol stacks that handle the requirements of applications while omitting what's not needed. The work centers around the use of a synchronous language called Esterel. Esterel allows protocols to be constructed from functional building blocks with explicit dependencies and state transitions. Using Esterel they hoped to be able to generate a number of very general-purpose building-block modules to enable new protocols to be designed and implemented quickly.

To validate their model, they implemented a TCP-like protocol using Esterel (they omitted the connection establishment and termination phases) that uses three major modules for send, receive and connection management. The Esterel code examples and module descriptions did look simple and elegant. However, the Esterel code was still 10,000 lines and the compiled code was 100KB; four times the size of the corresponding compiled BSD code. The large compiled code size was the result of no code sharing. Duplicate code in different states was duplicated in the compiled code. Additionally, they only achieved 1Mb/s using this protocol for file transfer. They determined that there wasn't any specific overhead in the generated C code, and the program flow was very similar to the hand-coded BSD stack. The problems were that a function call per Esterel program state was being generated, and the large code size wiped out the cache.

On the other hand, performance wasn't the goal at this stage. Their goal was to validate their model of flexible, modular protocol construction, which they feel they achieved. They believe that they can design application specific protocols which more closely match applications' needs and are more efficient than general purpose protocols. In their future work, they plan on working on some of the performance problems brought to light by this case study.

A Framework for the Non-Monolithic Implementation of Protocols in the x-kernel

Parag Jain, University of British Columbia

Parag presented his group's work on breaking protocol processing out of the kernel. They propose implementing protocols in user libraries with a flexible splitting of functionality between kernel and user code. The advantages include easier debugging of protocol implementations, applications-specific customizations and reduced-state contentions. The kernel minimally performs demultiplexing of incoming packets.

They have achieved performance very comparable to current stacks although connection setup is worse because shadow sessions have to be set up in the kernel to facilitate demultiplexing. They believe that this approach will scale up well on multi-CPU platforms because state data is private to the processes. There is some sharing of shadow-session connection state in the kernel but it's read-only.

Hugh Lauer decided to stir up trouble at the end of this session with the sarcastic remark that "both of these papers talked about performance in the Mb/s range. I thought we were at a symposium on high-performance networking. . . What should convince us that Gb/s performance is possible with either of these approaches?" Parag offered his belief that their work would parallelize very well. However, it was pointed out that this has yet to be proved in their work.

Device Drivers

Session chair: Tom Lyon, Mohr Davidow Ventures

Invited Talk: The AURORA Testbed

Jonathan Smith, University of Pennsylvania

Jonathan described the AURORA Testbed and discussed its status and future plans. The AURORA Testbed is a high-speed, wide area ATM network currently using STS-48 SONET (Synchronous Optical Network) links operating at 2.4Gb/s. The links are split into 3 STS-12 660Mb/s channels at each site. The links are supplied by three carriers: Bell Atlantic, MCI and Bellcore. Jonathan described some of the fun in getting the carriers coordinated on a project this size.

Some of AURORA's research goals are to investigate integrating different classes of service on a single network, (i.e., a network that's more than just a switching fabric), areas of applicability of different modes, (i.e., ATM versus PTM (packet transfer mode)), congestion control and management, and new routing services.

To support AURORA, they have built an ATM host interface for the IBM RISC System/6000 and integrated an IBM plaNET switch for the local switching architecture. Currently at the University of Arizona, where operating system support for Gb/s networking is being done for AURORA, they are achieving 90% of link speed for receiver processing, although sender processing is lower at around 300Mbs. (Jonathan mentioned why the difference but I seem to have lost his explanation in my notes.)

So far, the AURORA project has been very hardware oriented and is only just now addressing software applications. For example, one of the fun toys they've built is a $400 card that directly converts audio/video to and from ATM! With this card they perform remote learning experiments on a daily basis. They also have several other experiments running including a robot remote control system at the University of Pennsylvania.

Embedding High-Speed ATM in UNIX IP

Jonathan Smith, University of Pennsylvania

In this presentation Jonathan discussed the work going on at the University of Pennsylvania to implement high-performance ATM host interfaces, particularly the work necessary to integrate these interfaces into a traditional UNIX kernel (AIX) while maintaining high-performance. Their fundamental goal is high application-to-application throughput.

Key ideas in the OC-3c interface include splitting cell operations from data movement, doing ATM segmentation and reassembly (SAR) in hardware and providing selected support for demultiplexing in the reassembly process. The interface is implemented as a pair of cards: the segmenter and the reassembler. With older RS/6000 platforms, the bottleneck was the MCA-IOCC bus. Under newer platforms the bottleneck has become the OC-3c link itself.

New ideas they have explored on the software side include operating system structures for zero copies (ala the HP AfterBurner) and "clocked interrupts." The clocked interrupts are basically UNIX callouts implementing programmable polling. The effort here is to avoid a big interrupt hit for every small packet. Latency is a big issue with this, and they are looking at adaptive scheduling.

They have measured performance using their raw ATM interface at 130Mb/s without overloading the workstation! Even under load they were able to achieve continued high throughput at 83Mb/s. The performance for TCP is less good, but still a very respectable 63Mb/s for TCP transmission using ttcp.

In describing their experimental setup one of Jonathan's humorous observations concerned a very effective way they'd found to debug problems: by watching the LEDs on the interface boards. He recommends that hardware designers use lots of them. They're cheap, fast and effective at spotting key problems, much like the hardware equivalent of print statements. Also interesting: in their experiments they use a locally developed SONET NULL modem!

For their future work, Jonathan mentioned three areas of interest. In their current interface, segmentation and reassembly is done in interface memory, which effectively implements a store and forward network between the host and the interface. It may be valuable to perform segmentation and reassembly directly to and from host memory. They also want to experiment with long haul paths by setting up virtual circuits that snake back and forth across the WAN, and an HP Snake/AfterBurner implementation is in the works. (This paper failed to make it into the proceedings, but it is available via anonymous ftp: *ftp.cis.upenn.edu:pub/dsl/hispd.ps*).

Device Driver Issues in High-Performance Networking

John Tracy, University of Notre Dame

The fundamental thesis that John presented was that, as everything in the system speeds up, device drivers have to be careful not to become bottle necks. John and his coauthor Arindam Banerji's contention is that current driver architecture has remained mostly static while advances in link hardware, protocols, and APIs have advanced under pressure from new high-performance application domains. Their research goal is to advance device-driver analysis. The metric they used for their tests was round trip latency between two user-level clients. The array of performance-enhancing techniques they investigated was large. A brief list of some of the techniques which had a beneficial impact are:

- Use of fast paths. They coded optimized paths for special cases like multipacket send. The contention is that increasing functionality and generality in the driver adds undesirable branches in the driver code.

- Optimistic interrupt protection. The act of changing the interrupt priority level is expensive and rarely needed. They simply set a flag indicating non-interruptible status and defer interrupt processing to the end of normal processing. They provide a single level of deferred processing.

- Shared memory to reduce copying. There are several issues with this. For security reasons they need to implement "move" semantics. They also do separate per domain allocation (I presume to reduce contention and fragmentation). The implementation is via a shared heap kernel extension and a user level mbuf allocator. No other kernel modifications were required.

The total latency improvement gained after all modifications is about 9.5% on Ethernet and 18.5% on a 220Mb/s link. To me, this seems like a very modest improvement for a lot of complexity. But maybe the changes aren't as complex as they sound.

This talk engendered quite a few questions. Two notable ones were directed at John and his coauthor's decision not to contaminate driver code with upper layer protocol-specific information. One asked how much device-driver optimizations can be generalized? Another asked why not contaminate the driver? John answered that there are some common operations (like data movement) that need to be done, and these changes aren't restricted to UNIX. Also, if an implementor absolutely wants the last bit of performance, go ahead and contaminate the driver, but understand that contamination yields lots of problems.

John was also asked "what one thing would you ask from hardware designers?" John responded that he'd like to see interfaces that support single copy implementations with copying directly to and from user memory, scatter/gather, etc. Interestingly, he stated that he didn't feel that big interface memories are necessary. I'm curious how he intends to provide buffering between applications and the wire.

Performance

Session chair: Bill Johnston, Lawrence Berkeley Laboratory

Invited Talk: The CASA Testbed

Paul Messina, California Institute of Technology

Paul presented a fascinating view of the work being done on CASA. CASA consists of set of geographically distributed supercomputer centers locally networked with HIPPI (High Performance Parallel Interface) and linked together with fiber-optic SONET links. HIPPI frames are transparently encapsulated on the SONET links. (The HIPPI to SONET links are described in Kevin Fall's following paper.) The supercomputers include: Crays, a Thinking Machines CM5, massively parallel machines from Intel, and a host of other exotic hardware.

One of the goals of CASA is to be able to treat multiple supercomputers as a distributed computing cluster. To this end they are using Express which was developed by CalTech and evolved by Paragon. (I gather that Express is a set of software that provides an API to simplify distributed processing applications and hide the lower-level communication implementation details.) Current research activities include: tightly coupled data-parallel computation, functionally-distributed applications like atmospheric and ocean modeling, and distributed visualization and data fusion.

One application has yielded a key result: super-linear speed up. The application is a chemical-reaction dynamics problem using quantum analysis. The solution involves two major phases, SF and Log-D, which are O(N^3) in execution time. Currently problems of size N=1000 are being tackled. Super-linear speed up is achieved because the Crays are better at the SF phase and the CM5 and Intel machines are faster on the Log-D phase. Paul gave one example that showed how, when the problem is decomposed onto a Cray C90 and a 64 node Delta, the overall time to execute is about one quarter what it would take to execute the entire problem on either machine.

Another application involves fusing data from LandSat, seismic reflection and digital topography for the Colorado River extension corridor. It allows interactive browsing at several frames a second. They've also encountered the "Field of Dreams" effect: build it and they will come. Several new applications have sprung up that weren't envi-

sioned, for example, an adaptation of the SGI flight simulator to interactively fly through a dataset for Death Valley. The same was also done with data from the Venus mapping project!

TCP/IP and HIPPI Performance in the CASA Gigabit Testbed

Kevin Fall, San Diego Supercomputer Center

Kevin talked about CASA's performance experiments using HIPPI and TCP/IP. In particular Kevin is concerned with TCP's response to packet loss. Their experiences show that even low loss rates are interpreted by TCP as network congestion which tickles TCP's slow start algorithm. The result is that throughput goes to hell.

The CASA testbed consists of several HIPPI local area networks (LANs) connected via HIPPI to SONET Gateways (HSGs) developed at Los Alamos National Laboratory. The HSG provides local HIPPI termination to get by HIPPI connection timing constraints and buffering on the remote side to handle remote output blocking. Output blocking occurs in HIPPI when a HIPPI port is already receiving data. New requests for connections to that output port will be blocked until the current connection is closed, typically at the end of an application message. When the HSG encounters output blocking it can do a number of things: drop the message immediately, drop it after some specified time-out period or hold it indefinitely until the output blockage clears.

Kevin's report concerned itself with the case of HSGs dropping packets after a specified time-out period. Their investigation showed that TCP interpreted the losses as congestion which resulted in triggering TCP's slow start algorithm constantly. Kevin argued that such output blocking losses should not be interpreted by TCP as congestion.

I discussed this with Kevin extensively afterwards, and the two of us got a chance to talk to Van Jacobson for a few minutes on Tuesday morning. The fundamental problem is that when Van made his congestion avoidance and recovery changes to TCP back in 1988, TCP was already widely distributed. It simply wasn't feasible to change the protocol. So he changed the protocol processing instead. In particular, TCP doesn't have selective acknowledgments or negative acknowledgments, so Van was forced to interpret certain implied signals like ACK timer expirations. In his and others' experimentation on the Internet at that time, almost all lost packets were the result of congestion. Almost none were due to corrupted packets and lossy links. As a result, he chose to use missing ACKs as a congestion signal. The CASA network in Kevin's experiment didn't fit this assumption. Effectively it presented a lossy link!

Van also said that slow start maximally introduces a 25% performance drop in TCP on lossy links. He advocated introducing selective acknowledgments into TCP (something that RFC 1323, TCP Extensions for High Performance, specified; this was unfortunately dropped from the eventual extension specification). Hopefully, selective acknowledgments will find their way into TCP, probably the next generation TCP. As the Internet expands into wireless communications areas, lossy links will start proliferating.

Tuesday, August 2:

Keynote address: Learning from the Present-Things that IP got Right and ATM got Wrong

Van Jacobson, Lawrence Berkeley Laboratory

> "Q.93B (the ATM signaling standard) is ugly and foolish. It's a repeat of all the mistakes X.25 made. X.25 fell on its face and now we're trying it again. Meanwhile IP has been a tremendous success and over half of all current X.25 traffic is carried in IP encapsulation across the Internet."

With that as an opening (paraphrased) salvo, Van Jacobson launched into a fascinating hour and a half, point by point attack on ATM (Asynchronous Transfer Mode). However, it's important to note that Van isn't attacking all uses of ATM. He is attacking the belief that a vast, world-wide ATM cloud can be constructed with ATM virtual circuits replacing IP packets. In particular he had these good things to say about ATM: ATM is better than Time Domain Multiplexing (TDM) for high-speed trunking between Internet routers because it's cheaper and more flexible. Also, the bandwidth independence of the ATM interface standard is very useful for host and network interfaces. This should make it very good for high-speed LANs.

However, the Internet is not just hosts and trunks. In fact, IP explicitly doesn't care about interconnect technology, just that things are connected.

Van outlined what he feels is a classic hierarchy of networking problems that protocols must address if they are to be successful:

- Going fast,
- Getting big, and
- Crossing borders [between organizational entities].

These get harder from top to bottom: going fast is hard, crossing borders is within epsilon of impossible. IP is the first protocol to handle all these problems (although just barely in the last case). There are many experimental protocol which are fast for small N.

Getting big:

ATM is a bad match for everything we know about data traffic. Virtual circuits work when call lifetime is long compared to the call setup time. Unfortunately all Internet studies to date have found that the average connection size is somewhere between 2 and 5KB. For a 1Gb/s transcontinental pipe with 120ms round trip time (RTT), which means that call lifetime should be about 25 microseconds, but it gets inflated by a factor of 5000 because of setup time. This generates completely useless state in the network for 4000 times the average message transmission time!

There's one place where current Internet traffic is long lived compared to RTT: IP multicast backbone (MBONE) voice and video. Unfortunately, ATM's multicast model can't support this. In the IP multicast model, receivers announce interest, senders just send, and the network takes care of delivering data from senders to interested receivers. This scales by O(log R). It works because multicast addresses have global meaning and provide a network-level identity for the session. In the ATM multicast model everyone has to know everyone.

ATM has scaling reliability problems, too. ATM state is spread out over all switches in the path. The Virtual Path Identifier (VPI) and Virtual Circuit Identifier (VCI) in the ATM cell is per hop. If a hop's reliability is lambda, then the probability of failure for an N hop path is 1 – N*lambda. For a typical router reliabilities of 99.99% uptime, path failure for a typical 22 hop path is 1%. For moderately well connected IP networks, failure probability goes to 0 because of dynamic rerouting. IP separates network state from connection state. In other words: ATM fails if anything fails; IP fails if everything fails.

The central problem is that the core ATM element, the call, is not a low level building block like the IP datagram. It's a high-level abstraction.

Crossing borders

Crossing borders is really about making promises, part of an organization's willingness to carry transit traffic and thereby erase boundaries. An IP router's promise is very weak: I'll make a best-effort delivery. By contrast, an ATM switch's promise is very strong: I'll send cells on an established VC path, I won't crash and I'll remember your call until you hang up.

The key reason for boundaries is to control what and how much crosses them. Usually this involves a feedback loop with data and control flowing in opposite directions. ATM insists on hop-by-hop flow control, as if chaining pairs of nodes together is the same problem as a single pair. One (of many) problems with hop-by-hop flow control is that it exports problems in one part of the net into other parts (flow controlling a node causes it to flow control its inputs and so on). "Ringing" can spread throughout the network and become a stable phenomena. Hop-by-hop flow control is not intrinsic to ATM (ATM could use end-to-end flow control) but there is an odd desire for hop-by-hop!

In his concluding remarks, Van noted that ATM creates boundaries where they are not needed. In fact, everything is a boundary! (This is in reference to the ATM User Network Interface (UNI), Network Network Interfaces (NNI) found all over ATM network architectures). By contrast, nothing in IP is a boundary. By careful engineering and complex negotiation, it may be possible not to send data. If the object of this exercise is to communicate, IP works much better than ATM.

As one might guess, an address like the above is bound to stir some blood, and there were plenty of questions and objections. The most interesting point made was by a Fong Liaw from Fore who said that Fore has proposed a new ATM multicast modeled on IP multicast. The proposal has been accepted by the ATM Forum and Fore is in the process of writing up the specification. Van said he was very glad to hear this, but the rest of his objections remained.

During the break, another person asked why Van's complaints didn't apply equally as well to the current global telephone system since it operates on the same model that ATM does. Van responded that the global telephone network is run by a very small number of very large organizations who cooperate closely to maintain a very shallow network architecture. Most calls only go through three to five switches. Additionally, the telephone switches are massively over-engineered to make failure rates vanishingly small, which makes it a very expensive model.

Distributed computation 1

Session chair: Gerald Neufeld, University of British Columbia

Invited Talk: The MAGIC Testbed

Bill Johnston, Lawrence Berkeley Laboratory

MAGIC is another of the high-speed, wide-area testbeds that ARPA is funding. Its fundamental goal is to investigate interactive browsing of large datasets in a distributed, high-performance system. What Bill described was impressive. The size of the datasets and performance requirements are daunting.

Their current dataset consists of 1 meter/pixel imagery of Fort Irwin, CA. This data was created by scanning photos. Orthorectification of these images is very difficult because of inaccurate camera position knowledge. (Orthorectification is the process of creating orthographic images from aerial or space-based images.) The next generation of imagery will be 0.2 meter/pixel, will be captured digitally,

and camera positions will be tracked with GPS receivers. Orthographic image generation will be fully automatic.

SRI has created an application called TerraVision which provides reality-based navigation through landscapes combined with live information display. The live information comes from vehicles at Fort Irwin which all have Global Position System (GPS) receivers and are connected to the Internet. The application requires both steady state data traffic during roaming and large bursts when the view is "teleported." The bulk of the data is imagery. In most demanding situations (teleporting), hundreds of Mb/s are needed. TerraVision is implemented on a 4 CPU SGI Onyx. The visual simulation images are created by texture-mapping terrain imagery onto a three-dimensional digital terrain elevation (DTED) polygon mesh. Video clips of live TerraVision sessions can be accessed from the MAGIC WWW home page at *http://www.magic.net.*

The technology components behind this are crucial. One of the most significant elements is the Image Server System (ISS) which implements a (currently) read-only, distributed, parallel and scalable mass storage and retrieval system. In order to provide access to the stored information with low latency the data is stored in a very structured and hierarchical manner. The ISS' function is simple: it is a virtual block server. The network technology is equally important. MAGIC was initially planned with OC-12c ATM (622Mb/s). Currently the OC-12c link is split into 4 OC-3 interfaces (153Mb/s). The ISS automatically stripes data out over the four interfaces.

Other major issues for study include: IP over ATM, network congestion and its impact (the ISS makes a pretty good load generator) and network management (large scale and hierarchical down to the individual interface level).

Systems Issues in Implementing High-Speed Distributed Parallel Storage Systems

Brian Tierney, Lawrence Berkeley Laboratory

Brian described many of the system-level performance issues in designing and building the MAGIC ISS server described above. With a big emphasis on parallelism and using off-the-shelf parts for the system, the design goals were to make the ISS be scalable in all dimensions (dataset size, number of users and number of servers) and be affordable. The prototype ISS lives on four to five UNIX workstations, each with four to six fast SCSI channels and ATM interfaces.

The design approach for the ISS was to assemble a geographically distributed, fault tolerant system. To achieve the performance, they need the ISS to

- avoid memory copies wherever possible
- use data placement algorithms to spread adjacent tiles to separate disks, with semi-adjacent tiles close to each other on disks
- use path prediction to stage data-using techniques like branch prediction
- instrument all components of the system for timing and data flow (putting together a well-balanced system requires a lot of analysis).

Currently the ISS has been ported to Sun, DEC and SGI platforms and is achieving about 60Mb/s with a single ATM interface and 110Mb/s with an ATM plus an FDDI interface. Future work will focus on better performance, new functionality and the ability to write to the ISS.

A 1.2 Gbit/sec, 1 Microsecond Latency ATM Interface

Ron Minnich

Ron described MINI, a Memory Integrated Network Interface that implements distributed memory via standard SIMM slots. SIMMs are pulled out, the MINI interface plugged in and the SIMMs installed into the MINI interface. They chose this approach because most I/O busses are too slow and have high latencies. This eliminates the OS from most transactions. All resources are aligned on page boundaries and VM is used for access control. For sending an ATM cell the VCI is determined from the virtual address. Since no interrupts are available from the SIMM buss, some form of polling is required, like the AURORA optimistic interrupt scheme.

Also, the interface is not cache-coherent, so multi-CPU platforms either have to map the MINI memory uncached or make sure only one CPU at a time accesses a particular region of MINI memory. They've performed extensive simulation. Performance is limited to 0.966Gb/s because of HPGL chip processes. Hardware and software are under active development.

The development and testing process for MINI is very interesting. They're testing the hardware design using a VHDL simulation. To test application software, they've developed a C++ application programming interface that allows applications to communicate through the simulated MINI hardware in the Verilog simulator. This leads to a very interesting situation where they get the synergistic benefit of being able to test the MINI design with real applications and test those applications with a simulation of the real MINI interface. Of course, this could also make it hard to figure out where errors are coming from!

Distributed Computation 2

Session chair: Lixia Zhang, Xerox PARC

Invited Talk: The BLANCA Testbed and XUNET

Pat Parseghian, AT&T Bell Laboratories

XUNET is one of the oldest of the high-performance network testbeds. It's managed by AT&T Bell Laboratories and spans the continental US, from Murray Hill, New Jersey, to Berkeley, California. Most of the links are 45Mb/s with two now at 622Mb/s. Most of the XUNET sites are universities and XUNET's primary emphasis is on student research. As such, the entire network is reprogrammable down to cell handling. Research projects include studies on "fair" queuing models, congestion control, network management, quality of service (QOS) guarantees, etc. A XUNET student conference is held every year in early February to allow students to present their work. (I've been to one, it was interesting and fun.)

XTP as a Transport Protocol Support for Distributed Parallel Processing

Tim Strayer, Sandia National Laboratories, California

Tim reported on Sandia's benchmarking experiments for clustered computing communications. They believe that XTP best matches cluster computing requirement because XTP connections can negotiate only the functionality needed. In this case, they want reliable datagrams.

Their experimental testbed included Sun SPARC workstations running SUNOS 5.3 (*Ed: Solaris 2.3*) interconnected via a DEC FDDI Gigaswitch. They measured message latency for XTP, TCP, UDP and R-UDP (a homebrew reliable datagram facility built on top of UDP).

They determined that XTP was about 30% faster than TCP for one-shot delivery, but about the same as TCP when sending on a pre-established connection. Unfortunately, the time to send on a pre-established connection was well over ten times faster than XTP's one-shot time for small messages. Thus, they reluctantly concluded that pre-established connections should be used if at all possible. They also concluded that both TCP and XTP need to be able to set up, send and tear down connections faster to reduce one-shot and transaction message deliveries.

ViewStation Applications: Intelligent Video Processing Over a Broadband Local Area Network

Christopher Lindblad, MIT Laboratory for Computer Science

The one description for this talk has to be: cool! Christopher described ViewStation, a distributed multimedia system based on UNIX workstations and a Gb/s LAN. ViewStation applications are built using the ViewSystem, a toolbox to construct interesting and fun video processing applications, and run over the ViewNet, a home brew 4 port ATM crossbar switch. Some of the interesting applications they have developed include:

- The Room Monitor: a tool to watch a room through a camera and record the video any time there's movement.
- The News Browser: a tool that watches (television) network news and indexes it by keywords (using closed captions for the hearing impaired as keyword content source).
- The Joke Browser: a tool that watches David Letterman's opening monologue and segments and indexes the jokes by keyword.

A video browser is used to index all of the above databases. To see some demonstrations of their work see *http://tns-www.lcs.mit.edu/vs/demos.html.*

One of the more interesting results of their work relates to the implications of multimedia applications on network traffic: potential network traffic is a combination of bursty and periodic transfers, even for multimedia. Multimedia traffic is not all one-hour video streams!

ATM as a Networking Infrastructure

Session chair: Jeffrey Mogul, Digital Equipment Corporation. Session panelists: Fred Sammartino, Sun Microsystems & The ATM Forum, Steve Deering, Xerox PARC, Chuck Thacker, Digital Equipment Corporation

This was a face-off between an ATM enthusiast, Fred Sammartino, an ATM detractor, Steve Deering, and an ATM heretic, Chuck Thacker. Jeff Mogul said he felt he was qualified to moderate this since he hadn't yet formed an opinion on ATM. (I think we need to nominate Jeff for the Supreme Court).

Fred Sammartino opened by (humorously) projecting that the entire human population would be on the Internet by the year 2001 given current growth trends. He demonstrated his belief that people will send pictures instead of email in the future with the example of how the comet impacts on Jupiter affected the Internet. The Internet is also going real-time with all sorts of multimedia traffic. The implication is that we need a network infrastructure that can support the entire human population sending real-time videos to each other. (This is my interpretation of his lead in, not his!) Fred believes that ATM holds the answer for this problem because it can support virtual paths and circuits, handle resource reservation, as well as offer bandwidth and latency guarantees. He projected that widespread ATM deployment would reach the corporate backbone by 1995-96 and the desktop with multimedia by 1997-98. His current list of hot issues include: LAN emulation (ATM will be a failure if it can't be

integrated into installed networks), and congestion control among others I couldn't write down in time.

Steve Deering jumped off from Van Jacobson's morning keynote address by asserting that ubiquitous ATM is a dream (partly because of existing LAN investments), even when the end-to-end connection is ATM there is little benefit to "optimizing" IP, and eliminating the Internet layer is a Big Mistake. IP insulates applications from subnet technology: it works over heterogeneous subnets (even itself), allows subnet technologies to be deployed transparently, and provides a good least-common denominator. Replacing IP with ATM is a paradigm shift backwards: all currently successful networking technologies have been based on connectionless datagram models. In looking at using ATM as a subnet technology under IP, Steve noted that the Internet is evolving to support new classes of service, that it's likely the IPng (the next generation of the Internet Protocol) will be ATM's only payload traffic, and, even though IP can run over anything, ATM is almost the worst possible match to IP's requirements. So, what's good about ATM? It's switch based so we get greater aggregate bandwidth. But we were going that way anyway. It's high-speed (but would be even higher without packet shredders in the links!) Permanent Virtual Circuits might make good, flexible virtual wires.

Chuck Thacker titled his presentation "The Dark Side of ATM." He works on an ATM-like switch under development called the AN2. The AN2 has 16 800Mb/s ports. Each port holds a quad 155Mb/s or a single 622Mb/s SONET interface. The AN2 is an input-buffered crossbar with random access to prevent head-of-line blocking. His view of ATM problem areas include congestion control, LAN emulation, network availability and management, and the standards process. Chuck presented a case for Available Bit Rate (ABR) hop-by-hop flow control based on a servo-system analysis. (Unfortunately Van Jacobson wasn't there to debate this with Chuck. I think it would have been very interesting since they both are approaching the problem from a control systems standpoint.) Chuck also voiced a complaint about the ATM standards process: "we need to standardize things that work. Standards bodies like to invent things; they are reluctant to standardize things already in existence because of a perception of unfair advantage." Chuck feels that the standards process slowed when the ATM Forum came into existence.

There were many questions and long discussions after their initial presentations and follow ups (not covered here). I've run way over my space budget, so I won't repeat the common attacks and counter attacks in the ATM wars. But I do have to repeat one quote from Mike O'Dell which I thought was priceless: "The good thing about ATM will be that it will get SONET deployed and we'll have nice fast paths to use when people give up on ATM."

Summer '94 Conference Reports

Magic Considered Harmful: Coping with the Mysteries in a Modern UNIX Environment

Douglas P. Kingston <dpk@morgan.com> and
Daniel F. Fisher <dff@panix.com>
Reviewed by Peter Collinson <pc@hillside.co.uk>

This talk described "how to do better" when running production UNIX systems in an environment where things had better not crash or zillions of dollars will be lost.

In the last few years, this work has been done at Morgan Stanley, the bank. The problem is to maintain the operability of large network of machines running large complex applications. Most of the code is Third Party Software supplied in object code form only and there is often no system document ion.

Basically, things break or software rot sets in. The problem is what to do about that. Some problems are easy to solve but there are a number that are difficult. These still need fixing to ensure that the whole system remains stable.

You could just ignore the problem, but that doesn't make it go away. Also it means that you don't get experience with the problem, after all this problem may actually be solved. It's a bad idea to kill and restart processes: this just destroys the evidence. Rebooting is worse. It can cure things, but some level of stress on the system will probably bring it back. Also, rebooting is disruptive in the system and will hide resource leakage.

It's better to publicize the problem: let everyone know that you are looking at it because maybe someone else has solved it. Try and reproduce the problem by using small test cases or try to make it fail. Try to understand what the problem is, eliminate the impossible cases and gather evidence. Test your hypotheses, don't assume that your guesses are correct. Whenever a new problem is discovered make sure that you keep as much information as possible.

The talk then turned to a number of case studies where the available tools provided a diagnosis of the problem. These were interesting studies not only because of the use of available tools, but because they demonstrated the power of an organization that spends huge sums on hardware and software. A few of the problems were in publicly available software, but the majority were fixed by "leaning on the vendor."

I imagine that these problems usually result in patches. Somehow I feel we need a public-clearing house for "bugs that have been fixed for me by my vendor". However, it's clear that having cast-iron evidence of a bug helps in generating fixes from vendors. To get that evidence you need tools.

The talk mentioned a number of tools that have been helpful in problem detection. Good tools interface well with other UNIX programs. Their output should be parse-able if necessary. They should run without the requirement to set things up in advance. They should have minimal overhead or impact on the system.

Tools that are useful are:

top	To look at the most prominent system hogs
mon, monitor	To watch system status in realtime
pmon	To watch processes status in realtime
strace, trace, truss	To trace processes
tcpdump	To watch those packets fly by
lsof, ofiles	To look for processes that have named files open
traceroute	To follow that packet through the net
resource	To print resource usage data returned by system for a process on exit.
rcpinfo, netstat, pstat	To machine resource monitoring
gcore	to get a core dump of a running process, better than using SIGQUIT
adb,dbx	To look at bits and bytes
od	A UNIX Version 6 tool, still going strong
nfswatch	As it says
SIGSTOP	To suspend rogue processes for later investigation
strings	To get text strings from programs

Future issues include some new tools for new technologies like mapped files, and dealing with greater uses of distributed services, such as and mail and distributed naming. There is a need for more comprehensive interfaces to system information.

The talk's main message is that following up on problems and getting them fixed is critical to long-term system stability. Most instability is caused by just a few problems. It's easier that you might expect to fix these, and the payoff is larger than you probably expect.

Memory allocation

Dirk Grunwald & Ben Zorn University of Colorado at Boulder
Reviewed by Peter Collinson <pc@hillside.co.uk>

This concerns two related talks on memory allocation techniques. Both talks represent work that the speakers have been doing together for a considerable time.

Dirk spoke first. The team has spent time in examining algorithms for malloc and measuring their performance in several programs. Timing and space measurements show clearly that there are marked differences among the performances of many common allocators. The trade-off is generally speed against space. For example, the Sun code is the slowest, but it also uses the least space. Some programs spend as much as 25% of their time in malloc and free.

Dirk has constructed tools to examine the dynamic behavior of programs. He showed a screen-shot video of this in action. Looking at several programs, Dirk and Ben found that most tend to allocate and re-allocate objects of the same size. If you measure malloc calls, you don't get a standard distribution for the size requests.

Dirk then gave an overview of the different algorithms that are in common use. Most of them don't pay attention to the common behavior of allocating the same-size blocks again and again.

He spent time demonstrating that the coalescing blocks (adjacent small blocks joined into one big block) is not a good idea. It is better to keep the blocks split up in the size that was originally requested. When the program asks for a block of the same size again, you are ready and waiting to supply it.

Some algorithms worry about the paging behavior of the machines. It's not a neat trick to touch lots of pages in the program's virtual address space when freeing or allocating memory.

The main conclusion is that you need different memory allocators for different applications. Dirk and Ben have written a malloc called CustoMalloc that adapts to the application. The basic idea is to run the application program, collecting statistics on memory use, then generate a malloc algorithm depending on the numbers that are collected. It creates storage "classes" for the program based on the size of blocks that are requested. It uses a statistical profile to tune the memory allocation and optimizes common cases for malloc and free.

The measurements show that this approach works well to generate an optimum allocation method for the program.

The second talk, by Ben Zorn, examined how to make storage allocation "correct" in the sense that dynamic allocation is a huge source of program bugs. These bugs are often hard to detect and linger in programs waiting to cause problems.

The main bugs are memory leaks and freeing a free area. There are other memory-related errors like reading or writing data outside an allocated area, accessing data through a null pointer or accessing uninitialised data.

There are two choices in getting around this. The first is to fix the code and find the bugs. The second is to adopt the approach that lisp uses, do garbage collection.

To find bugs, a variety of approaches exist. You can use a version of malloc that is instrumented. It allows you to see what is happening with the program. You can use an allocation profiler like mprof that pinpoints the responsibility of block allocation and frees. Finally, you can use a system that instruments the code itself, like Purify.

Another approach is to forget about the de-allocation of space and use garbage collection to retrieve unused areas of memory. Basically the data area of the program is searched for valid pointers and the blocks that are addresses are marked. All other blocks can then be discarded. The problem with this approach is that it's hard to "know" what a memory value is. Something that looks like a memory address might not be one. For this reason, garbage collectors for C and C++ are known as "conservative." Garbage collection costs CPU time and memory to find the unused blocks. Also, you have to pause doing the "application" and do some housekeeping.

However, garbage collection avoids leaks, since it sweeps up the debris and avoids freeing free blocks. Ben has been looking at a publicly available Garbage Collector implemented by Hans J.-Boehm, Alan Demers, and Mark Weiser at Xerox PARC. He has done a study looking at CPU overhead, memory overhead, reference locality and ease of use. He has compared the Boehm-Wieser collector against four efficient malloc/free implementations. He's used eleven C and C++ programs for testing.

On the whole, you pay for the ease of programming that garbage collection allows. The CPU overhead is application-dependent. Generally, garbage collection is 20% slower than the fastest malloc. Garbage collection needs 1.5 to 2.5 more memory and it has worse locality of reference. On the bright side, you can plug it in easily.

Ben ended by saying that correct dynamic storage is difficult. A number of debugging tools exist and many errors can be eliminated. Conservative garbage collection solves some problems, the CPU and memory cost may be reasonable.

The authors intend to put their notes and many references for further reading up for World Wide Web consumption, see *http://www.cs.colorado.edu/homes/zorn/public_html/DSA.html*

The 25 Years of UNIX panel session

Reviewed by Peter Collinson
<pc@hillside.co.uk>

It has probably not escaped your notice that the Boston conference celebrated 25 years of the UNIX Operating System. Peter Salus, author of the recent book on the early history, had arranged a couple of panel sessions. I volunteered to try to cover the first of these for *;login:*.

On the panel were Armando Stettner, who was responsible for raising the profile of UNIX within DEC; Mel Ferentz, who was responsible (with others) for the start of the USENIX Association – he generated the first newsletter "UNIX NEWS" that became *;login:*; John Lions, who was responsible for documenting the Version 6 kernel in a way that made it accessible to a great many people – it's probably the most photocopied pair of books in the world; finally, Dennis Ritchie, who was, well, just responsible.

Mike O'Dell kicked off the session as moderator with a tale of his first attempt to get the UNIX system. He was a student in Oklahoma and came across the CACM paper. This made him somewhat excited. The university had a PDP-9 and well, maybe that would run the system. So he called Bell Labs and asked to speak to Dennis. He had anticipated layers of secretarial defense and was highly surprised to find himself speaking to Dennis. Dennis persuaded him that the code would not run on the PDP-9 and that was that. He went off to get a PDP-11.

Dennis then had the floor. He commented on the Salus book and said that he felt that some things should be put straight. First, he wanted to correct the business of the name. The book says that Brian Kernighan had thought of the name UNICS as a pun on Multics. "Well", said Dennis, "that's not what I remember, is it Brian?" There then followed an exchange which completely muddied the waters. I spoke to Brian afterwards and he said that Dennis recalled that Peter Wienberger had thought of the name, while Brian remembered that he did. So . . . we are all no wiser. We shall never know. It will be one of those great unsolved mysteries.

Second, Dennis wanted to remark on the famous and oft-quoted kernel comment

```
/* You are not expected to understand this */
```

Dennis pointed out that this was not a bald statement standing on its own; it was the last line of a seven-line comment. The preceding statements tried to explain what was happening. The "you are not" line was intended to be reassuring. The code implemented the process-context switch and

depended on the procedure-calling sequence in the C compiler. However, in retrospect, neither Ken nor Dennis understood it. The code was wrong.

Mel Ferentz then spoke. He, too, recalled the experience of getting the system out of Bell Labs. He called and asked for Ken. He got through to Ken's secretary who said that "no, she didn't know whether Ken was in, and no, she couldn't find him because the UNIX room was miles away down the hallway, and anyway he probably wasn't in." Mel should try at lunchtime. All the programmers arrived to have lunch, and he could talk to Ken then. (Dennis said that this no longer happens, the UNIX room gets doughnuts brought in.)

Mel dealt with the lawyers at AT&T and got them to insert a clause in the license allowing people with licenses to talk to each other, the consenting adults clause. This clause permitted Harvard to print the first set of manuals and sowed the seeds of USENIX.

John Lions talked about the early days in Australia. He mentioned Greg Rose and Chris Maltby who got the University of New South Wales system running on a PDP-11. George Coulouris asked how much work it took to do the books. The books were conceived as aids for student teaching. The book containing the listing of the code was done one year, and the notes that described the code the next. John was too modest about the effort that it must have taken to understand and annotate the almost-uncommented code of the kernel.

Armando talked about early days in DEC. He had been responsible for raising the profile of UNIX within DEC. Later he laid the directions for Digital's early native VAX UNIX products and lead the development team creating VAX Ultrix, V1.0. He said that when he and Bill Shannon set up decvax, the first Usenet hub, it probably had the largest phone bill of any computer in the country.

In addition to recounting his early days at Digital, Armando also recounted his first encounters with UNIX while a consultant in New York and later, while at working at Bell Labs. He spoke of the trouble that he had in explaining to his mother that he didn't need a surgical operation before going to a "eunuchs" conference.

There were a number of questions for the panel from the floor.

Dennis was asked why the Seventh Edition went out with the Bourne shell and not the Mashey shell. The Mashey shell had been developed to support PWB UNIX. This system acted as a batch processing front end for big IBM machines and was supported by a number of scripts in the Mashey shell. The shell was an optimized version of the V6 shell: it had test as a built-in command, for example. This was not developed inside Research but in a different arm of the AT&T organization.

Bourne had developed his shell in the UNIX group in research. He had come from the University of Cambridge in England and wanted to use Algol 68 rather than C. The syntax of the Bourne shell came from Algol 68 and the code was written in a dialect of C looking like Algol 68. This was done using macros in the C pre-processor. The debugger, adb, was also written in this language. The "a" in adb stood for "Algol."

Anyway, Mashey realized that the Bourne shell had more future as a language, it had better control constructs. He adopted it for PWB UNIX.

Dennis was asked why the "for" statement in the Bourne shell ends with "done". There is "if" and "fi," "case' and "esac," why not "do" and "od." Simple. Steve did implement this but people complained when they could not use the "od" command.

Dennis was asked what he would do differently if given the chance. He made the usual joke about `creat`. He said that he wished they had gotten onto trends earlier. For example, they didn't pick-up on screen editors. There were also many little mistakes . . . had the `stty` definition changed earlier, there would have been less confusion between the System V and the BSD (evolved Version 7) way of doing things.

All in all, this was an enjoyable session. It ended with storytelling from the floor. It's still a strange thing that people love UNIX so much. You don't really see anyone get misty eyed about early MS-DOS, do you.

On the Internet, Nobody Knows You're a Program

Rob Pike
Reviewed by Peter Collinson
<pc@hillside.co.uk>

This was the funniest talk of the entire conference, surpassing Penn Jillette's presentation. People walked out of this one visibly aching from laughing so much. Rob tells me that he intends to write this up "properly" so I will refrain from spoiling too many of his excellent jokes.

Rob talked about his several experiments in the early '80s, when he realised that he could derive great amusement from playing with the Usenet news. His basic thought was to write a program that would automatically create and submit news articles. He never really got there, but came close.

He "came out" as the creator of net.suicide and admitted to being the horrendous Bimmler, named after a Monty Python character. Bimmler was vicious, had no thoughts of suicide, was a good writer, attacked everyone for anything, often

because they had odd names. She just attacked people. Bimmler was seriously nasty, a true Hell's Granny. She was born to instant infamy by taking over net.suicide in a *Coup d' Etat*. In fact, nearly everyone on net.suicide were creations. Rob stirred things up by co-operating with Ron Hardin, a.k.a. Chuck Festoon.

In the end, Rob felt that net.suicide was taking too much of his time and killed the group with a suicide note.

But Rob still wanted a program. He invented Mark V. Shaney, a character whose language was derived using a program written by Don Mitchell implementing Markov chains. The basic algorithm was to take some text, compute the probability of all three word sequences, and use the first two words. The output was generated based on a probability table. Punctuation was retained to improve readability. The programming was done by Bruce Ellis.

It worked. The program generates comprehensible incomprehension. The stuff reads brilliantly but is entirely meaningless. Rob took net.singles for a day and put the text through the program. He was allowed to rearrange sentences; he could fix punctuation to improve meaning but could not edit words or sentences.

He posted the following article, the first of several:

```
From alice!mvs Sept 1984

I'm saying that Christians recognize sex
within marriage as the only proper sphere of
sex, and the backs of their appearance.  Why?
Because first impressions are important.
Because appearance has subtle effects on
mood.  Because it is part of the September 84
edition!  800 pages, mostly advertising.

I deal with it by simply not letting it
bother me.  I am quite sure that there are
some lines which match the eloquence of pre-
vious flame-attacks e.g. something like "Cap-
italist Running-Dog Pig-F***kers", a classic
of the way a person man or woman looks during
sexual climax: red lips, flushed cheeks, red
purple eye lids.

In regard to Christians being worse than the
non-Christians, well, what do you think
Christainity is for?  Paul the Apostle spoke
of this "Man-Catching Bust in 90 days" crap.

Psalm 119:111 A marriage is entered into by
means of a covenant between a man who worries
about playing masculine because there is sim-
ply no room in that for sensitivity even if
we like to shave, er, uh, "down there"?  As
for the kids?  It doesn't.

When I meet someone on a professional basis,
I want them to shave their arms.  While at a
conference a few weeks back, I spent an
interesting evening with a grain of salt.  I
wouldn't take them seriously!  This brings me
back to the brash people who dare others to
do so or not.  I love a good flame argument,
probably more than anyone...

I am going to introduce a new topic: does
anyone have any suggestions? Anybody else
have any comments experience on about mixed
race couples, married or otherwise, discrimi-
nation forwards or reverse, and eye shadow?
This is probably the origin of makeup, though
it is worth reading, let alone judge another
person for reading it or not?  Ye gods!

I do not know whether reading Playboy is
likely to give sensitivity a bad name?  For
people who suffer from high muscle tension
i.e., are screwed up, it's a great relaxant
and anesthetic.  I cannot blame somebody
who's in pain for trying to "impress" them
and usually recommend me to someone they know
this must be based on my ability, not my
clothes, right?

I have chuq down as a technique if not a prin-
ciple.

Mark
```

Operating System Support for Distributed Multimedia

Reviewed by Jerry Peek
<jerry@ora.com>

The Pegasus project is building a general-purpose distributed multimedia system. The components are connected by an ATM network through a DAN (desk-area network) switch. The DAN switch isn't a crossbar but rather a highly parallel setup that can even look like a bus; more switches can be added to get more bandwidth. A camera, display and DSP node are connected directly to the network – not through a workstation bus. The ATM camera produces digital video, a stream of ATM cells, from a frame buffer. Video can be compressed in real time with JPEG. Video is encoded as 8-by-8 pixel *tiles*. Instead of "bursty" frame-by-frame data, tiles are sent immediately; there's no frame delay. The OS is *Nemesis*, a microkernel with some unusual features for multimedia applica-tions, see the paper in the Proceedings. A UNIX server "on the side" can create, control, and communicate with Nemesis processes. All time-critical operations are done in hardware.

The most interesting part of the design (for me) was the way that devices ("ATM devices") are connected directly to the network. Some highlights:

- The CPU isn't slowed down as a side effect.
- ATM cameras and displays are simple and potentially cheap.
- The setup integrates well with existing ATM systems and networks. Existing OS software doesn't need to be modified.
- ATM should become more standardized than processor buses, so a design like this will be usable everywhere.

One of the applications being developed is a digital TV director. It can be used to show a meeting remotely over a slow network. The camera is controlled by calculating the voice position from microphones and automatically aiming the camera. It isn't working perfectly, but it's a good demo of a user-level application.

Splicing UNIX into a Genome Mapping Laboratory: How we subverted a Mac interface and programs to be front-end components of a UNIX toolkit

Lincoln Stein
Reviewed by Jerry Peek
<jerry@ora.com>

The presenter and his team provide computer support for a group of biologists. The biologists, who are doing some 2.5 million experiments as part of a genome mapping project, are used to a Macintosh interface. They have a lot of trouble adapting to Motif (they use a one-button mouse, they ask things like "Why don't cut and paste work right?"etc.). But the Mac environment wasn't great for the huge data sets and constantly-evolving research and analysis methods.

After a series of trials (including Smalltalk and from-scratch Mac GUIs), Lincoln's team finally decided to use Mac spreadsheet, email and TCP/IP to interface with a UNIX system. The UNIX system runs a combination of Perl scripts and *cron* jobs to do the number crunching. The system works well; the biologists can use their Macs and the programmers can hack the flexible UNIX setup.

The biologists never access their database directly:

- To submit large amounts of data, they drop a Mac file into a folder that's connected to UNIX with NFS. A Perl script, run by *cron*, crunches any data in the folder every half-hour.
- To submit tabular data, an Excel spreadsheet with custom macros (such as "Send to database") is used. The spreadsheet transmits requests by a TCP/IP socket to a waiting Perl script which reads the tab-separated data and sends a response through the socket.
- Database queries are handled with an email system: Microsoft Mail and its email forms. Microsoft Mail creates the text-format message that UNIX expects – and sends it with SMTP. Email works well here because the processing delay (20-30 seconds) is too long for an interactive setup but very acceptable for email.
- The nightly *cron* jobs crunch data and email a status report to the scientists.

You can get more information and a copy of the paper from *http://www-genome.wi.mit.edu* or by FTP from *genome.wi.mit.edu* in the directories */distribution/stein* and */distribution/steve.*

Works in Progress Reports

by Lou Katz, Permanent Past President
<lou@orange.metron.com>

Peg Schafer chaired this even dozen short reports on current work.

Cache File System in Solaris 2

Neil Groundwater
<Neil.Groundwater@central.sun.com>

Solaris 23 has integrated a CacheFS in order to accelerate access to NFS and CDROM file systems. CacheFS can improve the utilization of network bandwidth and server/disk access such that more "clients" can be supported by the same resources. Use of CacheFS is transparent to NFS servers, an advantage in heterogeneous networks. After a quick overview, this talk will give some instances of usage and performance measurements. Current work involves "CacheOS" where CacheFS is used to cache all of Solaris. CacheOS client machines become field replaceable units with small local disks. A white paper is available at: *http://www.sun.com/sunsoft/Products/Solaris-whitepapers/Solaris-whitepapers.html*

Crashme, Experiencing Random Stress

George J. Carrette
<gjc@mitech.com>

The author describes his experiences and that of his internetwork correspondents in adapting a user-mode-instruction random-input testing tool to the latest RISC hardware and software architectures.

The program tests the robustness of an operating environment using a technique of "Random Input" response analysis, such as formally proposed in the book *Cybernetics* by Norbert Wiener. However, any parent who has

observed his children playing and learning would be well disposed to describe this in detail.

Results for a range of hardware and software architectures, including ALPHA, PA-RISC, and SPARC are summarized.

Security and the World Wide Web

David I. Dalva
<dave@tis.com>

Lots of organizations want to run Mosaic. Many of these, especially the ones protected by firewalls, are concerned about exposing their internal networks by letting HTTP through the firewall. HTTP proxies are available that are designed to run on a firewall system and make the Web transparent from the inside. Are these proxies enough to protect internal systems from unauthorized access?

This talk will describe several security problems that exist in today's methods of access to the Web, and will present proposed solutions to some of these problems. (*http://www.tis.com/Home/NetworkSecurity/www.html)*

A Network Monitor Approach to Security

Craig Leres and Vern Paxson, Network Research Group, Lawrence Berkeley Laboratory <leres@ee.lbl.gov, vern@ee.lbl.gov>

We've been working on using a packet-filter-based network monitoring tool to trace our site's wide-area network traffic. Scripts periodically reduce the traces to one-line summaries of each connection, which are then analyzed for patterns indicating possible security attacks, including: excessive use of finger, excessive rejected connections, ftp connections, busy host pairs, IP address spoofing, and outside access to key machines such as gateways, nameservers, network monitors, and key servers. The scripts are also valuable in building up a profile of the site's general network use, and in the process have uncovered a number of previously undetected network problems.

swIPe this disk! Release 1.0 of swIPe

John Ioannidis
<ji@tla.org>

swIPe is a network-layer security protocol for IP; it has been described in a paper at the 4th Usenix Security Symposium (John Ioannidis and Matt Blaze, "The Architecture and Implementation of Network-Level Security Under Unix" in *Proceedings of the 4th Usenix Security Symposium*, Santa Clara, CA, October 1993), and an Internet Draft (John Ioannidis and Matt Blaze, "The swIPe IP Security Protocol," Internet Draft, December 1993). A *free* implementation of swIPe is now available as a modload-able Sun 4.1.x device driver. This talk will describe the swIPe implementation, some applications, and tell you how to get your very own copy.

Software packages from the Network Research Group at Lawrence Berkeley Laboratory

Craig Leres
<leres@ee.lbl.gov>

This talk is a brief overview of the software packages available via anonymous ftp from the Network Research Group at the Lawrence Berkeley Laboratory. Some of the more well known packages covered are:

- CSLIP (Compressed Serial Line IP),
- FLEX (Fast LEXical analyzer generator),
- TCPDUMP (protocol packet capture and dump program), and
- VAT (Visual Audio Tool)

These and more can be found in the top level directory of *ftp.ee.lbl.gov (128.3.112.20).* Don't forget to set binary mode before retrieving the compressed (*.Z) files!

Fetch-news

David Muir Sharnoff
<muir@idiom.berkeley.ca.us>

Fetch-news is part of my ongoing quest to have a useable internet link over a modem. Fetch-news is a news transfer agent. It figures out what news to fetch by reading `.newsrc` files. It pulls the news by pretending to be a newsreader. Unlike some similar programs, fetch-news uses article numbers to figure out what to fetch. News is injected using the `ihave`. In combination with a tcp wrapper that limits the send and receive buffer sizes, fetch-news is able to provide timely and complete news service without destroying the interactive responce of a SLIP line.
URL: *ftp:://ftp.idiom.com/pub/muir-programs/fetch-news.*

Performance Modeling of the DCE RPC

Peter Honeyman
<honey@citi.umich.edu>

(Peter Honeyman gave one of his briefest talks to date . . .) We have developed an analytic model of the response time for the non-idempotent form of DCE RPC. We decomposed the DCE RPC into stages and measured the time consumed in each stage as precisely as possible. The time spent in a particular stage can be regarded as the service time for a hardware device associated with that stage. To account for queueing delay, we use a layered queueing model that uses the measured service demands as model parameters.

File Layout and File System Performance

Keith A. Smith Harvard University
<kieth@das.harvard.edu>

Most contemporary inplementations of the Berkeley Fast File System (e.g. those found in BSD 4.4, SVR4.2 and SunOS) attempt to optimize file system throughput by clustering data from the same file on physically contiguous disk blocks on the file system. Over time, and with higher disk utilization, one would expect the free space on a file system to become fragmented, limiting the ability of the disk allocation algorithm to effectively cluster file data. In order to examine this issue, we have collected data about the file layout on our file servers over a six month period. Some of the questions we are attempting to answer include: How successful is the disk allocation heuristic? How does file system performance change with time? Are older file systems slower than new ones? How does file layout change for diffrent types of files (large files vs. small files, old files vs. new files, etc.)?

Cicer: A Package Installation System for an Integrated Computing Environment

David Bianco
<bianco@rave.larc.nasa.gov>

This WIP describes Cicero, a system for installing and managing software packages across a large heterogeneous network. This system is being developed at NASA's Langley Research Center (LaRC) in Hampton, Virginia by members of the Integrated Computing Environment (ICE) team. Although there are many different aspects of ICE, this paper focuses on the ICE team's work in progress concerning the issue of software distribution over a large installed base of clients. (*http://ice-www.larc.nasa.gov/ICE/doc/Cicero/cicero.html*)

The SNOBOL4 Language Shares Some History With Un*x

Phil Budne
<budd@cs.bu.edu>

Both were implemented at Bell Labs, with portability figuring strongly into the implementations. Both were ported early in development, and later ported to many systems. Interaction between the two groups is responsible for similar features in AWK and SNOBOL4. Unlike Un*x, SNOBOL4 code has always been freely available, and it not widely used today.

I have done a public domain port of SNOBOL4 for Un*x.

Design of a Video Storage System

Pen-jen Oyang

The presentation is about the design of a video storage system for on-demand playback. The design stresses effective utilization of disk bandwith with minimal data buffer to minimize overall system costs. The system is most distinctive in a novel data placement strategy that is based on a two-level hierarchical disk array architecture. If compared with other multimedia storage system designs that have been proposed for the same applications, the system proposed in the presentation enjoys a distinct advantage of overall system costs.
(*ftp:// earth.csie.ntu.edu.tw:/pub/techreports*)

USENIX C++ Conference Report

by Susan E. Waggoner Shopiro, NationsBank and Jonathan Shopiro, Merrill Lynch

The Sixth USENIX C++ Conference was held at the Cambridge, MA Marriot Hotel on April 11-14, 1994. Tutorials were given on Monday and Tuesday; technical sessions, BoF meetings, and a vendor display were on Wednesday and Thursday, and an Advanced Topics Workshop on Distributed Programming was held on Friday at the Boston Computer Museum.

The keynote speech of the conference was given by L. Peter Deutsch, designer and implementer of a variety of languages and systems, and co-recipient of the 1993 ACM Software System Award. His renown is such that when he was late to the Advanced Topics Workshop, Jim Waldo worried that he had been captured by the computer museum staff for use as an exhibit! It is traditional for keynote speakers to remind us of our shortcomings, and Peter did not disappoint.

Deutsch's talk, "C++: A better C – for whom?" characterized C++ as an experiment that has been too successful. He described a programming language "food chain" from language designers to compiler and tool implementers to users (programmers). He suggested that there should be separate languages for system and application programming. The apparent goal of C++ of being suitable for all programming tasks has led to its being too low-level and dangerous for application programmers, too full of hidden performance penalties for system programmers, and too large and complicated for everybody. He faulted the design process that led to C++, arguing that a language should be completely designed before it is implemented and unleashed upon the world and that compatibility with previous versions was incompatible with good language design. He suggested that it was time to look at the results of the C++ experiment and design a new language for

system programming, extending C with just a few of the most useful and best-defined features of C++. He even presented a chart suggesting which features should and should not be included.

The technical sessions on Wednesday and Thursday included 17 papers and a panel on experience with C++ in large systems. The session topics were extensibility, compilation, debugging, memory management, design, and tools. The proceedings of the conference, containing all the technical papers,are available from USENIX.[*See order form on page 000*] Most of the papers focussed on ways to increase the usefulness of C++ rather than suggesting changes to the language itself.

C++ has been recommended as a language for writing reusable code but it is a common experience to find that code is not so reusable after all. The papers in the extensibility session uniformly advocated improving the separation between interface and implementation as an approach for producing more extensible code, although they proposed different ways to do it. Vince Russo and his students at Purdue have been working for several years on "signatures," a language extension that permits implementation-free interface specifications, and Vince and Gerald Baumgartner presented a paper describing not just one implementation of signatures, but three. Other people have proposed using virtual base classes and pure virtual functions for the same purpose, but performance problems have limited the popularity of this approach. A paper by Lee Nackman and John Barton of IBM explains this programming technique and gives a method for compiling such programs that should improve their performance.

One of the primary bugaboos of C++ programmers is memory management, and there have been attempts to address this problem from the early days of the language. Two papers on garbage collection for C++ were presented at the conference. One, by John Ellis of Xerox and David Detlefs of DEC, has been creating quite a stir circulating on the net for a while and seems to be closer to practicality than any of its predecessors. The other, by Giuseppe Attardi and Tito Flagella of the University of Pisa, describes how several different approaches to memory management (including Ellis and Detlefs') can be combined in the same program.

Debugging was the primary topic of papers by Jim Coplien of AT&T and Chris Laffra and Ashok Malhotra of IBM. Cope advocated object- rather than class-based breakpoints, so that the program pauses only when a member of a particular object is invoked. The IBM authors described HotWire, a C++ visualizer that allows the programmer to watch the growth and transformation of his or her data structures.

The papers in the Tools session focussed on collection and use of information about C++ classes. It will be interesting to follow the progress of these ideas as compilers that support run-time type information become widely available. Then much of the information that is now gathered by specialized analysis programs will be easily available as the program runs.

And now a word from our sponsors: the members of the panel on using C++ in large systems were Wendell Baker from the Berkeley CAD group, Kevin Brown from Morgan Stanley, and David Goldsmith from Taligent. A common theme from the panel was that the complexity of the language was a hindrance. The compilers available to them are inconsistent in what language features they implement and in how they implement them. They expressed concern over the vagueness of the language definition and urged the Standards committees to work hard. They did not, however, advocate removing any language features, in fact their large systems need and use more of the language features, making support more challenging. Also, they found that the development tools available don't scale up to large systems. The panel members said that finding experienced OO developers was still difficult which meant that many engineers on the project would be novices and need extra mentoring. They discovered the need for organizational standards and enforcement of them. Cowboys beware!

Despite these problems, the panel agreed that the advantages of C++ were worth the headaches. As Goldsmith concluded, "C++ is the worst programming language for use on large systems, except for all the others. (Apologies to Winston Churchill)."

While the Berkeley and Taligent projects were presented as pure C++ systems, the Morgan Stanley Equities Trader's Desk application was a different thing entirely. It incorporates "legacy" systems written in non-OO languages and relational databases. These systems were given C++ wrappers and used as the models in instances of the "Model-View-Controller" pattern. Reliable interprocess communication was achieved using the ISIS system. This approach allowed them to present information to their users using the power of the OO paradigm, yet let them continue to use their finely tuned current systems. They felt this was a more conservative and practical approach than rewriting everything from scratch all at once.

The Advanced Topics Workshop was devoted to an advanced and hot topic, using C++ for distributed programming. The attendees were primarily concerned with the Object Management Group (OMG) and its Common Object Request Broker Architecture (CORBA), the 800-pound gorilla of the distributed programming world, but there were speakers who had gone their own way, both at the workshop and the main conference as well. Among them were John Dilley of HP who described his C++ interface to OSF DCE, and Daniel Edelson and Steve Williams of IA Corporation who described their C++ library, InterStage, for building distributed systems.

C++ library, InterStage, for building distributed systems. Christopher Pettus of Taligent described the distributed object service within TalOS, which is implemented in C++. Also Doug Schmidt of UC Irvine had a paper in the conference on a framework for distributed applications based on his previous C++ Wrappers work.

The CORBA specification includes an interface description language (IDL). Since CORBA-IDL is not C++, some effort is necessary to use IDL interfaces from C++ or to provide C++ implementations for such interfaces. The semantic gap between CORBA-IDL and C++ was a concern for several workshop participants. IDL has different fundamental data types (for example, sequence) than C++ and also IDL has a different runtime type information capability than C+. Philippe Gautron et al. of Novell, Chorus Systèmes, and Siemens AG gave a workshop talk on an extension of C++ with IDL features. Brian Lewis of Sun described the use of C++ and IDL in the Spring OS and in the Navigator, a Spring application. Ann Wollrath, also of Sun, described an effort to translate IDL to C++ that actually wound up translating IDL to Modula-3!

A conclusion from the conference is that while C++ is the language that people most love to hate, it is the dominant language for today and its successor is not yet on the horizon.

USENIX Supports Student Participation and Professional Development

The USENIX Association sponsors an Educational Outreach Program for faculty members of computer science departments around the world. These faculty liaisons provide their students access to USENIX publications, upcoming events as well as conference scholarships for full-time students. For more information on this program please contact Diane DeMartini at 1 510 528 8649 or <*diane@usenix.org*>.

A $500 cash prize is awarded for the Best Student Paper at USENIX annual Winter and Summer Conferences. (Students are eligible also for Best Paper and other awards.)

Membership in USENIX for full-time Students is only $20. As a member, your benefits include:

- Free subscription to *;login:*, bimonthly newsletter with articles, community news, calendar of events, book reviews, Standards Activities Reports, Systems Administrators Guild News, and more
- Free subscription to *Computing Systems*, refereed technical quarterly published with The MIT Press
- $75 conference registration fees (vs fees of $275 to $340 for as many as ten technical conferences and symposia each year),
- $50 tutorial fees,
- Discounts on conference/symposia proceedings, other technical publications, and books from the new series on advanced computing published with The MIT Press,
- Eligibility to join SAGE, the Systems Administrators Guild,
- And, maybe most importanty, participation in leadership in the systems community.

For membership information please contact the USENIX office by calling 510/528-8649 or sending email to <*office@usenix.org*>.

Community News

Tina Darmohray had her baby on Sunday, August 21st. His name is Craig Francis Marker, he was born at 11:51 A.M., weighed 9 lbs. 4 ozs., and was 21 1/2 long at birth. Tina and Craig are doing fine.

Please send news and notes of happenings (marriages, births, deaths, moves) for publication in this column to <*login@usenix.org*>.

SAGE, the System Administrators Guild, is dedicated to the advancement of system administration as a profession. In just two years, SAGE's membership continues to increase steadily, and there is growing recognition of SAGE as a representative in system administration issues. SAGE brings together system and network administrators for:

- professional and technical development,
- sharing of problems and solutions,
- communicating with users, management, and vendors on system administration topics.

SAGE NEWS EDITOR
- Bryan McDonald <*bigmac@usenix.org*>

SAGE BOARD OF DIRECTORS
- Elizabeth Zwicky, President <*zwicky@usenix.org*>
- Paul Evans, Secretary <*ple@usenix.org*>
- Peg Schafer, Treasurer <*peg@usenix.org*>
- Paul Moriarty <*pmm@usenix.org*>
- Pat Parseghian <*pep@usenix.org*>
- Steve Simmons <*scs@usenix.org*>
- Pat Wilson <*paw@usenix.org*>

SAGE WORKING GROUPS

GROUP	CHAIR
sage-certify	Arch Mott
sage-edu	Paul Evans
sage-ethics	Ed Gould
sage-jobs	Tina Darmohray
sage-locals	Rene Gobeyn
sage-online	Mark Verber
sage-policies	Lee Damon
sage-pubs	Bryan McDonald
sage-security	Arnold & Laura de Leon
sage-stds	Janet Frazier
sage-vendors	Dave England

You can contact these groups via email at <*their name@usenix.org*> for example, <*sage-certify@usenix.org*>.

SAGE DISCUSSION GROUPS
sage-managers
sage-outreach
sage-pt

SAGE ONLINE SERVICES
Email server:
majordomo@usenix.org

FTP server:
ftp.sage.usenix.org

WWW URL:
http://www.sage.usenix.org

SAGE SUPPORTING MEMBER
Enterprise Systems Management Corp.

SAGE
THE SYSTEM ADMINISTRATORS GUILD
NEWS

On Professionalism

by Tom Limoncelli
<tal@plts.org>

Elizabeth Zwicky <*zwicky@corp.sgi.com*> writes about the value of going to a LISA conference:

> If you've never had the experience of being in the same space with dozens, hundreds, or thousands of other system administrators, you owe it to yourself to try it; even if the tutorials and the talks don't help you, the discussion at coffee breaks is an education in itself.

This is an excellent lead-in to something I've been wanting to bring up for a while.

What is the difference between an amateur and a professional? I'm not talking about whether or not you are paid, but whether or not you act amateurish or professionally and whether you are treated with the appropriate amount of respect by the people around you, the people you work for, and society.

At a recent retreat, a friend brought up the fact that in her field (psychiatry) extra schooling is required to become professionalized. What she discovered was that 70% of her professionalization was the result of being with other people going through the same "professionalization" process. Growing together, maturing together, watching their teachers, they all learned how to walk, talk, dress, act like a "professional." By the time their schooling was done, the psychiatrists were "professionals" (the social definition, at least).

Hearing that really got me to thinking. One of the goals of SAGE is to get more respect for system administration as a profession.

Yet, we have no schools that let us interact with others for years before we hit the "real world." In fact, it is quite the opposite. I think the term "on the job training" must have been created to describe system administrators.

System administrators usually work in isolation. There may be one per department, one per area, etc. If you are the system administrator for a bunch of chemists at a medical research company, there is no one you can turn to and ask for technical opinions or even talk about better ways to manage your time, do major overhauls, etc. The nearest thing you have is Usenet, which many system administrators don't have access to. Even if you do, it can be difficult to explain an entire 10-net, 100-machine network so that you can ask, "What's the best way to configure automounter in this situation?" If you do, you're likely to get 50 replies saying, "Ugh, I hate automounter. It sucks, dude!" Other technical questions might be difficult to ask because you are skirting exposing proprietary information.

So, what do we system administrators have instead?

We have LISA/SAGE/SANS/USENIX conferences. They are our universities where we interact with others, learn from scholars and get sage advice.

These conferences go a long way to reaching the goals of SAGE. However, they are only a couple of times a year, not everyone can attend, they are expensive and require absence from work for long periods of time.

$GROUPNAME is an organization in New Jersey (the birthplace of UNIX) which is like SAGE-AU or SAGE-UK. (*$groupname is not like SAGE-AU or SAGE-UK; it's a local group, like BayLISA and BackBAy LISA – Ed.)* We host "cluster groups" every other month. We pick restaurants around the state, (one in the south, one in the center, etc.) and we hold social gatherings simultaneously. Sometimes we pick a theme and report back what each group did with the theme. For example, once we had each group come up with a list of what was their most used publicly-available software. We all learned from each other at that meeting. It was interesting to compare what the groups listed in common. [1]

The cluster groups provide a social space for system administrators. Some people model good behavior while others learn. Sometimes it's like "Usenet-live" with ten people talking at once about how they switched configured automounter, or the talk gravitates to meta-topics like how to manage time, users, managers, etc.

How difficult is it to form a cluster group? Not very. We gain members by posting to the appropriate mailing lists and newsgroups. We do all the arrangements via email. The most important thing is to have consistent locations (so people only get lost their first time) and consistent time/date (so people can plan for it). Best of all, there can't be any turf wars. The hope is to have cluster groups going on in each part of the state. If a turf war starts, start an additional cluster!

What about the future?

Well, I don't think that cluster groups and conferences alone will change our industry. The SAGE Job Descriptions Booklet, salary surveys, and other tools are also needed. Maybe someday it will be expected that system administrators will have some kind of advanced degree. Then again, the most accurate way to predict the future is to be the one that creates it.

If you are interested in starting a local group, the friendly people on the sage-locals mailing list would love to help you get started! If you are in New Jersey, you can get more information about $GROUPNAME by writing to *groupname-info@plts.org.*

---

1 To see the results of these discussions:
echo get groupname 94-04-21-central-jersey-\ cluster-report | mail majordomo@plts.org
echo get groupname 94-04-21-south-jersey-\cluster-report | mail majordomo@plts.org

Surviving Solaris 2.x

by Hal Pomeranz
<pomeranz@netcom.com>

We Have Met the Enemy

If it hasn't happened to you by now, it will soon: somebody is going to hand you a SPARC machine and ask you to install Solaris 2.x on it. I'm part of a team of folks at my company who are rolling out a new computing infrastructure which is based on Solaris 2.3 (and NIS+, just to complete my happiness). Let me tell you now, there is no point in arguing: Sun is committed, the people who pay your salary are committed, and failing to jump on the technology early is going to be one of those career limiting decisions that you will regret in about two years.

Sure the paradigm shift is painful. And sure, I will admit to being a closet System V admin: I spent my formative system administration years at Bell Labs (but I was primarily a SunOS admin, and probably the only `csh` user, so I wasn't entirely brainwashed). There are, however, several things that you can do to make your transition less painful.

Patch! Patch! Patch!

While the stability of the operating system has increased rapidly with each successive point release, Solaris 2.3 right off the operating system media is pretty shaky. Get hold of your Sun rep and get Sun to provide you with a list of the "recommended" patches and install them. If this means paying for a support contract, then do it, even for just a year. Compare the price of the contract against the downtime you will experience without the patches and you will be way ahead.

Sun has also released a "Maintenance Supplement" CD. Volume 1 contains about five dozen patches that have all been QA'd together. Most of the patches have been superseded by newer versions, but it is a fine place to start and hopefully Sun will produce a second volume with newer patches. Sun's part number for the CD is SMSP-230-CDB-MS1.

Avoid at all costs patch 101318-50 which will leave your system in an unbootable state. I have heard, but not confirmed, reports of problems with other revs of the 101318 patch above -50. I am currently using 101318-45 without problems.

Third-Party Tools

You may have heard that you can eat anything if you just put enough ketchup on it. Well, you can survive any new operating system as long as you can get your favorite editor, mail agent, news reader, and windowing system all working.

Well the good news is that almost all of the "major" free software has been ported to Solaris 2.x. The GNU stuff is trivial; Perl works; there is a patch for X11R5 on ftp.x.org, and the list goes on. If you get stuck, send me email – it feels like I have ported every software package available on the Internet at this point (except building C-news or INN – our news server is a SunOS box – so you are on your own there).

And, by the way, I am tired of people whining about Sun unbundling the compiler: the war is over and we lost. You can ftp the Solaris gcc binaries from any GNU archive (and then do yourself a favor and use the compiler to build the latest rev of gcc from source). Actually, it is probably worthwhile to buy one license of the Sun unbundled compiler just to have it.

The first hint when compiling anything under Solaris 2.3 is to avoid the Berkeley compatibility libraries at all costs. The following macros should solve 90% of all of your problems in this area:

```
#ifdef SYSV
#define index          strchr
#define rindex         strnchr

#define bcopy(s,d,l)   memmove(d,s,l)
#define bzero(s,l)     memset(s,0,l)
#define bcmp           memcmp
#endif /* SYSV */
```

If it ends up that you simply must compile something `-lucb` (be sure to send me email for help first), be sure to also link it -Bstatic so your users do not have to add /usr/ucblib to their LD_LIBRARY_PATH variable.

Programs which use socket calls (almost anything vaguely distributed or network related) will need to be linked with "`-lsocket -lnsl`". Remember that this is a System V system, so there is no ranlib . Your best bet is to define a RANLIB variable at the top of your Makefile which you can set to "/bin/true" on System V machines and to "ranlib" on BSD boxes. Software which does not have a predefined configuration for Solaris 2.x will usually have a SVR4 or Systen V configuration predefined – use SVR4 if you can.

Just Do It

Learning a new operating system is just like learning a new language: total immersion is the best way to learn it. Make sure the display on your desktop is connected to a Solaris box. It will be painful at first and you will be tempted to switch back, but give it an honest try for at least a month after you have got your favorite tools ported. If you are still unable to deal with Solaris after a month, at least you will have some ammunition to fire at the enemy camp.

Do not try to turn on all of the whiz-bang features of Solaris at once. Wait until you have got the basics of System V system administration down before you start turning on things like cachefs and NIS+. In fact, wait for NIS+ to get a lot more "cooked" before you turn it on at all, or at least only enable it on a small workgroup at first so you can get the hang of it. You must apply the appropriate patches before enabling NIS+.

One caution: do not try to run Solaris 2.x on anything less than a SPARC Classic with 24MB of RAM – it is just too slow on weaker machines.

Why Ask Why?

There are plenty of reasons why you would want to go to Solaris 2.x, but everybody's reasons are different and I refuse to start a religious war on the issue. Sun is already shipping hardware that only supports Solaris 2.x, and somebody high up in your company is already planning on buying some of it. You can expend some effort now and dive into this stuff, or you can explain to your management later why nobody is getting any work done on their new $250,000 server.

Your Voice

by Win Bent
<whb@skat.usc.edu>

In previous issues of *;login:*, Elizabeth Zwicky has written about several "tools" she uses, and other aspects of system administration. Some of these have been extremely useful – her description of the trials of locating the power switch on a large variety of machines was perfect, except that apparently she was not confronted by an equal number of UPS boxes beeping at her.

However, I get a sense of the curmudgeon about her, especially after reading her two contributions to the June issue. She is correct that an improved understanding of the job of SAs will help all around, but when I read, "the real cost of work often doesn't get explained adequately to users who then believe that you are refusing to do something that is going to take three weeks to set up and three weeks a year to maintain because you're too lazy to spend five minutes to make them happy," I hear "everybody's picking on me." Her other column describes ways to hide from the users.

She poses, and answers, three questions concerning system administration:

- How much work is it really? More than the users think.
- How much work can the system administrator do? Less than the users think.
- Does the system administrator have motivation for doing new or difficult things? Usually not.

She mentions that the first is a communications problem, and the second is a problem of both communications and management. She doesn't say, but it's safe to assume that the third is a management problem. Nowhere do I see suggestions for solutions, and in particular her discussion of question three contains a fair amount of doom and gloom.

My favorite tool for system administration is my voice. If I am handed a project by someone who thinks it's merely a five-minute job, I make sure to let them know my view of it, either right away or in a follow-up call or e-mail. I let the users know where it stands in my priority list, and find out how high a priority it is for them. Yes, sometimes I am taken by surprise – an easy software installation turns out to be a major headache. Or vice versa! KEEP THE USERS INFORMED! They will have a better understanding of what you do and why it takes the time it does IF YOU TELL THEM ABOUT IT. "I had hoped to install the latest release of SpiffyMail by the end of today, but it turns out that the vendor sent us the wrong release on a sandpapered CD-ROM. It looks as though fixing this will take at least two weeks."

Similarly, you can create a level of motivation, or at least interest, in your management for new or difficult things. Tell them what a neat idea it is, how it'll be challenging, and how wonderful things will be when it's done. If they know what you're doing, and why you're doing it, they're much more likely to encourage you, and to recognize the achievement when the job is done.

A larger question is: what really is your job? Perhaps a very real part of your job is to pound non-stop on "five-minute" jobs. Perhaps you really are expected to say "No" to (almost) all requests, and perhaps you're even expected to do so without ever saying "Leave me alone!" Now's the time to use your voice: sit down with your manager and make sure you both understand what you're expected to do, and make sure you both understand how closely that matches the current reality.

Ann Landers often says, "nobody can take advantage of you without your permission." I believe this is true, if we allow for a wide interpretation of "permission." If you have become overworked and under-appreciated, it is up to you to change your situation. Talk to your users. Talk to your management. When everyone understands what you do and what you are expected to do, you and they will have come that much closer to being a team, not adversaries.

System Administration Models

by Elizabeth Zwicky
<zwicky@corp.sgi.com>

The most popular system administration model is none whatsoever. You have computers, some of which have system administrators. Which ones have system administrators is determined entirely at the whim of whoever purchases the computer. This model has the advantage of extreme simplicity, requiring no thought and no agreement to implement. On the other hand, it has severe disadvantages. It maximizes the extent to which decisions about computing are made on an ad-hoc basis to meet short-term needs. People buy computers that they can't use because there's nobody to help them set them up; people buy computers without buying backup systems or using them, and important projects are lost; resources are duplicated unnecessarily. This constitutes a major drain on corporate resources – most obviously, money, but most importantly the time and energy of people who are fighting computers instead of using them.

It is therefore in the corporate interest to actually ensure that there are system administrators and that those system administrators are effective. To that end, there should be some vision as to how those system administrators are distributed.

For some reason, the model that leaps to people's mind first is a purely centralized model based on the datacenters that became common in the days when IBM was still revising their estimate of the total number of computers the world needed from 6 to a few thousand. This inspires fear and loathing, and well it should; centralized datacenters were a neccessary evil, not a desireable solution, even when they were standard. In a world of workstations and personal computers, they are not merely inadvisable as a complete solution, they're impossible. If you attempt to centralize control of all the company's computing resources, issues of resource contention and local customizations will drive people to go back to buying their own unsupported computers, leaving you somewhat worse off than if you had just let them do it in the first place, since the new computers will be actively hidden. Nobody creates new purely centralized facilities, and the old ones discovered this problem with a vengeance when personal computers first became popular. This is therefore a straw-man argument, frequently mentioned, but only to be discredited.

In reality, a reasonable model will balance out a number of conflicting needs. You need to achieve economies of scale, by putting resources as high up in the organization as practical. You also want to standardize across groups in order to maximize your ability to move information and people

within the company. You need to custom fit the computers to their users as much as possible – one-size-fits-all works no better for computers than for clothes. You need to provide effective assistance to people, which means both having people who know what their particular circumstances and needs are and having people who are always available. You need system administrators who are responsible to the people they support, but those system administrators also need specialized management and support.

The end result is going to be some combination of centralized and local support. The centralized support is going to specialize in consistency, availability, and deep understanding of system administration issues, including explaining managers to system administrators and vice versa. The local support is going to specialize in local adaptation, user support, and representing the special needs of their community to the centralized support. Within this basic theme, there are many possible variations, mostly representing different solutions to two main problems: who is the line manager for the local system administrators, and what organizational level do they work for?

The local system administrators can either work for the central organization and be assigned to areas, or they can work for the local groups and be advised by the central organization. I admit that theoretically they could work for both simultaneously, resulting in an organizational directed graph rather than an organizational tree, but I sincerely hope that most businesses will reject this solution on the grounds that it is widely known to be a quick route to managerial disasters.

Having the local system administrators work for the central organization adds flexibility, helps to ensure consistency in both technical and managerial procedures, and ensures a support structure for the system administrators. On the other hand, it loosens the connection between the system administrators and the people they support, by making the system administrators into effectively internal contractors. (This is a special problem in companies that don't already have people in this sort of position.) In a company that is currently relatively strongly centralized, it may serve to increase local control; in a company that is currently very distributed, it will increase central control, which is likely to be a political and social problem. Whatever the organizational chart says, it will also give the system administrators a strong sense of having two managers, since they are trying to please both their manager, and the people in the local area they support. Again, this problem is likely to be most acute in companies where this sort of situation is rare.

Having the local system administrators work for the groups that they support avoids many managerial and political problems, at the expense of putting hiring and managerial decisions for system administrators in the hands of people who may not have the expertise in the are that the central group has. This tends to lead to inconsistent management, and encourages inconsistent technical decisions as well, by loosening the connections between the local groups. It also will leave most system administrators in very small groups (often consisting of lone system administrators). Some things cannot be managed at that level – networks, for instance – and must be provided as central services. Some things, like backups and security, are so crucial to the company that they must be monitored by a central group. Some things, like news and electronic mail, only provide cost-effective and user-friendly service if there is some centralized service available for them. Users need coverage when their system administrators are ill or otherwise absent; they need a reliable and easy-to-remember way to contact the correct people when something doesn't work, which almost certainly means a central dispatching service which can provide answers to routine queries, route problems to the appropriate person, and cover more hours than the local system administrators. Small system administration groups cannot be reasonably expected to be expert in everything and will need technical support in some areas.

All of this means that there will be a large number of problems and processes that involve both central system administrators and local system administrators, often in multiple parts of the central group and multiple local groups. These situations are extremely failure-prone, because of the number of people and hand-offs involved, and very hard to resolve when they fail, because there is rarely a single person at fault and a single place to fix things. This is by no means unique to the situation where local system administrators work for the local group, but it is significantly worsened there, because it increases the number of handoffs between different management trees. Handing a process off to someone with a different set of priorities and a different manager involves more risk of failure that handing it off within the same organization.

It is not unusual for companies to choose to combine these methods by leaving local system administrators managed by local groups where there are existing pools of administrators and expertise, and putting system administrators with central managers into the remaining groups. This is an effective solution for places that either want central control, or have groups that don't want to put the effort into hiring their own system administrators. Moving system administrators that are currently in local groups where they are well-integrated and effective is spectacularly difficult and unpopular. The opposite case – where the company wants to put in local system administrators but not everybody wants them — is less frequent and usually easier to fix. It usually results from a group that doesn't really want any system administrators at all, and the issues surrounding that will have to be dealt

with under either method. Mixing the two methods gives both sets of advantages, but it also provides both sets of disadvantages, and while each local group gets one set of advantages or the other, the central group gets to deal with both sets of disadvantages simultaneously.

All of this discussion has conveniently ignored the question of exactly what a local group is and exactly where the central group is. A local group needs to be a set of people with a strong common goal and culture, that already works as a reasonably cohesive unit. It needs to be large enough to support at least one system administrator, which leads inexorably into discussions of how many users a single system administrator should support. Most companies will want the largest possible number of people covered per administrator, which is partly a technical issue and partly a social issue. The technical issue can be roughly summed up by saying that the number of administrators you need rises with the complexity of the site and with the amount of support the users need. It is a mistake to assume that technically oriented users like engineers automatically need less support than non-technical users like secretaries; it is more accurate to say that they need different sorts of support. The social issue involves familiarity. The point of putting system administrators in local groups is to make them a working part of the group. This is not going to work unless the people who're being supported all know the system administrator. As a generalization, a local group should be no larger than a building, to encourage this sort of familiarity.

Picking local group sizes is extremely dependent on the corporate culture, and usually on the internal structure of individual pieces of the company. It's rarely possible to pick a single level of the organizational chart and declare it to be the right place to put the system administrators; not only does this usually leave local groups of extremely uneven size, it also leaves everybody above that level unsupported. This process should be regarded as an art, not a science, and left to fall out of the natural way the company works to the maximum extent possible.

The central group will generally be at the corporate level (although there is nothing stopping an individual part of the company from adopting this sort of model internally, when the company as a whole has no model at all, in which case the "central" group will be at the top level available). This is appropriate because issues of security, backups, and consistent electronic interface to the outside world should be corporate priorities, and because it leads to the maximum possible economies of scale. On the other hand, it may turn out to be reasonable to have sub-centers in smaller pieces of the company (at the risk of increasing all the communications problems). In particular, it's important to arrange financial matters so that purchases can be made where they are sensible. For instance, it may be reasonable for a number of local groups to share a file server that is not large enough for the whole company to share, and there should in that case be a mechanism for buying and supporting that machine. A programming division will probably have multiple local groups that share a need for the same compilers and tools, making it reasonable to buy and support those at the division level. It is in the best interest of the company to make it as easy as possible for groups to band together for site licenses of software and support; many companies either overpurchase software because this isn't possible, or end up wasting most of their savings in paying employees to argue about the financials. Ideally, these sub-groups can be handled in the center, but if that is not effective, putting in small amounts of intermediate organization may well be worth the effort.

All of this falls short of suggesting a particular solution for everybody's company. This is because there is no one right answer for the general case, although there may be a right answer for some specific cases. Almost any answer is better than ignoring the issue altogether, however; a traditional company large enough to have more than one building is going to need both local and central administrators, and they are going to need to work closely together.

Upcoming in *Computing Systems (Volume 7:4)*

ELK:
The Extension Language Kit
Carsten Bormann and Oliver Laumann

Secure Timeliness: On the Cost of Non-Synchronized Clocks
Raphael Yahalom

Reliable Real-Time Garbage Collection in C++
Kelvin Nilsen

Back issues of the journal are available at the following prices: $9 for individuals and $18 for institutions. Outside the US add $5 per issue for postage and handling. Canadians add an additional 7% GST. Contact the publisher directly for ordering back issues:

The MIT Press Journals
55 Hayward Street
Cambridge, MA 02142
1 617 253 2889
<journals-orders@mit.edu>

FEATURE

Understanding UNIX Workstation Performance

by Scott Hazen Mueller
<scott@zorch.sf-bay.org>

Introduction

UNIX performance and analysis and tuning is rather like the weather: Lots of people talk about it, but few do anything about it. Traditionally, it is the exclusive realm of people dealing in high-performance computing, either in corporate data centers or in scientific supercomputing environments. If you use a UNIX workstation, the usual response to a real or perceived performance problem is to run out and buy the latest and greatest fast box.

Why worry about performance?

I care about workstation performance for one of two reasons:

1) I can't buy a faster machine.

2) I don't think I should spend the money on a faster machine.

For either case, I try to measure the system performance and either verify or deny the statement, "This system's basic CPU performance and architecture is adequate to the task."

If I can verify this statement, then I try to identify changes I can make to the system that will allow it to perform up to capacity. If I deny it, I have to replace the machine.

Basics of system performance analysis

System performance analysis is really about the identification of bottlenecks, factors that prevent the system from performing the task(s) required of it. There are two kinds of bottlenecks:

1) Those that can be corrected.

2) Those that cannot be corrected without replacing the system.

Bottlenecks that cannot be corrected are those that are intrinsic to the design of the machine: the CPU type and speed (usually), the memory architecture, the I/O and I/O bus architecture, and operating system limitations. The only way to alleviate these bottlenecks is to replace the system, and sometimes to go to an entirely different platform. If, for example, your job mix requires that you run multiple parallel processing-intensive jobs, you may not be able to use any single-processor system. For that job mix, you probably have to use a Symmetric Multiprocessing (SMP) system.

Bottlenecks that can be corrected are generally resource bottlenecks: physical memory, disk I/O rates, and (sometimes) total CPU cycles. You can usually add memory to a system, for example, if you determine that you don't have enough. Disk loads can sometimes be split, balanced or striped across multiple disks. If you already are running an SMP-capable system, you can add more CPUs if you are running compute-intensive jobs.

Measuring system performance

In this section I'll talk entirely about the BSD-style tools, not because it's not possible to measure system performance on a System V system, but because I don't have access to a System V box to review the tools available there. Also, the discussion and analysis here is generic; it may not apply in a given situation because of special considerations. I hope to illuminate this topic, I don't claim to be able to solve everyone's performance problems.

My first tool is usually `ps(1)`. On a BSD (SunOS 4 also) system, "`ps -aux`" yields information on the memory and CPU utilization of currently-running processes. For example, from the output in Figure 1 I can determine that the system being used is largely idle, since "`ps`" itself is the process with the highest %CPU. The "SZ" column tells me the virtual (real + swap) size of the each process, and the "RSS" (Resident Set Size) information tells me how much real memory each process is consuming.

"`ps`" is also the best source of memory utilization information in most environments, and especially so on SunOS 4 systems. It breaks out the memory usage for each process, and the values can be summed to determine how much memory is being used by processes on the system. It does not tell how much memory is used by the kernel or for file system buffers. That information is actually fairly hard to obtain, and the method is highly system-dependent.

Use `vmstat(1)` for a broader overview of system performance. Do be aware that `vmstat` does not provide good numbers for systems with large numbers of disks, and that `vmstat` on SunOS 4 systems does not give good numbers for "avm" (Active Virtual Memory).

The output from `vmstat` (see Figure 2) at first glance appears to be quite cryptic, but it just takes a little time to become accustomed to the headers and to learn to decode it. The default display gives information on five areas: the process queue, memory utilization, paging activity, fault rates (interrupts and context switches) and CPU activity.

The process queue information is the information that underlies the (in)famous "load average" figure. If you have ever wondered how load average is computed, `vmstat` can give you the raw numbers. Basically, load average is the average number of jobs each second that are ready to run and waiting for the CPU. The "r" column under "procs" is the number of jobs this last second that are ready to run. Thus, for the example output, the load average would be about 1. The "b" column is the count of processes blocked waiting for I/O to complete, and the "w" column is the tem. A large count of processes ready to run may indicate that the CPU cycles available are not sufficient. Likewise, a large number of blocked processes could well mean that the system is waiting on I/O activity, and a large number of processes swapped out means the system probably doesn't have enough physical memory.

The memory information gives the "avm" (Active Virtual Memory), how much virtual memory is currently in use, and the size of the Free List ("fre"), basically how much physical memory is not in use. This information displays in a nutshell whether the system's physical memory is fully used or not.

The paging activity information gets quite detailed. Generally the "re" (Reclaim), "at" (Attach), "fr" (Pages Freed) and "de" (Deficit) figures are fairly meaningless unless you are actually debugging the virtual memory code in the kernel. The key indicators are the "pi" (Page In), "po" (Page Out) and "sr" (Scan Rate) numbers. The paging rates indicate, of course, whether the system is busy reading and writing pages to and from disk rather than doing real work. Even more important is the scan rate; BSD UNIX systems use a `clock' algorithm to search for memory pages to free, and the scan rate is the rate at which the system is performing free page searches. All BSD systems, as far as I know, are configured with a maximum value for the scan rate, two or three hundred on the ones I am familiar with. If the scan rate is significant, over one hundred, the system can be characterized as "thrashing", spending too much time on memory management. A system with a high scan rate needs more physical memory; this is one of the few calls I feel comfortable making without actually seeing the specific system.

Figure 1:

```
USER     PID  %CPU %MEM SZ   RSS  TT   STAT TIME   COMMAND
scott   2545  7.7  4.7  220  180  co   R    0:00   ps aux
root     143  0.0  17.4 716  696  ?    I    55:56  /usr/lib/news/trn/mthreads -s
```

Figure 2:

```
 procs memory        page       faults      cpu
r b w avm  fre  reatpipofrdesrd0d1d2d3insycsussyid
1 0 0 400  1908 0 0 0 0 0 0 0 0 0 0 0 4 6 1 0 1 99
```

The fault rate information gives an overview of disk activity, as well as the interrupt, system call and context switch rates. The disk activity information is critical in identifying disk I/O bottlenecks. Many systems will run disk I/O up to 30 I/Os per second (I am starting to see much higher numbers on newer systems with fast SCSI drives), so I/O rates in the 20 to 30 range are frequently indicative of a disk being driven flat-out. Since information is given for up to 4 disks, it is also possible to determine if the I/O load is well-balanced on a smaller system. A multiple-disk system should not have 30+ I/Os/second on one or two drives and <10 I/Os/second on the remainder.

The interrupt rate will be higher on systems with interrupt-driven I/O such as serial lines. The best advice I can give with regard to these figures is to generate some output under normal load conditions and then save it aside for comparison when looking at or for performance problems.

The CPU activity column is quite important. For one, if the "id" (Idle Time) figure is large, then it's almost a sure guarantee that your system can handle the workload and that CPU speed is not your performance problem. A low idle time, however, does not mean that the CPU is not adequate. It's entirely possible for a system that is thrashing to consume most of the CPU without performing any productive work. The CPU columns are displayed last if you read left-to-right, and it's a good idea to look at them last. Look for performance problems in other areas first, and use the CPU activity to verify or deny that your system has adequate performance.

There are other tools for delving in more detail into system performance; iostat, for example can give you all kinds of I/O activity information, and can be used to examine activity on specific disks. I believe that some systems provide a "`dkstat`" command, which provides information specifically on disk activity. The "`pstat`" command dumps various system tables, and can be used to examine the allocation of your swap space.

Improving system performance

You improve system performance by examining in turn each of the bottlenecks that you can alleviate, and for each that is a problem area you take the necessary corrective action. Always keep in mind the principle that correcting one bottleneck means that something else becomes the new bottleneck. If none of the correctable bottlenecks are causing your performance problem, you then have the option of either living with the problem, or replacing the system. I cannot give you good advice on how to choose a new system, except that you should study the design carefully and be sure that you understand the new bottlenecks.

Some key indicators that you might look for are:

- Lack of physical memory – high page out rates, high scan rates, high CPU idle time and/or high CPU system time, low free memory, large number of swapped processes.
- Disk overloaded – high disk I/O rates, high CPU idle time, sometimes high paging rates.
- CPU overloaded – no apparent memory or disk problems, low CPU idle time.

Frequently, the most reliable way to improve system performance is to add more physical memory. Adding memory can even improve performance on a system that is suffering apparent disk overload problems. If the system is indicating high disk I/O rates combined with high paging rates and/or a large number of swapped processes, the system is actually more short on memory than it is lacking disk I/O capacity. Adding more memory will lower the paging and swapping activity, all of which hits the disk(s), and will free up the disks for actual file I/O activity. Furthermore, on many systems adding memory will increase the size of the file system buffer cache, allowing more disk blocks to reside in memory, and decreasing the disk activity still more.

If you have no memory problems and disk activity is quite high, you can usually buy more performance in one of two ways. First, on a small system, you can move your data files to a spindle separate from the operating system itself, so that your data accesses don't have to contend for disk I/O bandwidth with OS activity. Secondly, on moderate-sized to large systems, you can use a virtual disk facility (for example OnLine DiskSuite from Sun) to aggregate disks into large partitions. "Stripe" disks together for better performance. Do make sure that you don't stripe disks together that are on the same controller.

Finally, if you have adequate memory and you don't have disk I/O problems, and your CPU is always busy, you may be able to improve performance by upgrading the CPU or adding another one. There are CPU upgrades available for some systems that are quite reasonably priced, simple to install, and that nearly double the raw CPU performance

Conclusion

It can often pay off quite handsomely to analyze the performance of a slow system before making the decision to replace it. Frequently, slow real performance is caused by a lack of resources rather than native lack of performance. Furthermore, taking the time to understand the performance of a system can let you take short-term measures to improve it in a "crunch" situation.

What's New

EDUPAGE
<edupage@ifovy.educom.com>

Study Urges More Pilot Projects. A study released June 14 by the Computer and Business Equipment Manufacturers Association and co-sponsored by Educom says research and development of the national information infrastructure should be guided by the results of pilot projects to test and evaluate technologies. Copies are available for $25 from Educom by faxing a request to: NII Forum, (202) 872-4318. (*BNA Daily Report for Executives* 6/15/94 A19)

Social Guide to Networking. It's no secret that networking happens both on and off-line, and the social aspects of working together through technology should not be overlooked. MecklerWeb President Chris Locke suggests seven principles of effective networking: talk to anyone about anything, develop a high tolerance for ambiguity, be willing to look stupid, give more than you take, cultivate fearlessness, go on gut instinct, and expand your sense of humor. (*InformationWeek* 6/20/94 p.94)

Computer Threat Shuts Down Mail Lists. Many Internet mailing lists that use Majordomo software were shut down for a few days this month following an alert issued by the Computer Emergency Response Team. CERT recommends replacing the 1.91 version of Majordomo with 1.92, which is available at: *ftp.greatcircle.com*. Then go to the directory */pub/majordomo* and retrieve a file called *majordomo1.92.tar.z*. (*Chronicle of Higher Education* 6/22/94 A21)

GTE Sets Up Interactive Video Unit. GTE Interactive Media is a new 80-person unit set up by GTE to create interactive video games and multimedia programs. At this week's Consumer Electronics Show in Chicago it will demo 20 new multimedia offerings, including games, educational programs for preschoolers, and a group called "virtual worlds" that mimic such things as a high-speed drive or a martial arts contest. (*New York Times* 6/21/94 C4)

Apple's Eworld Debuts. Apple Computer has launched eWorld, its on-line information service, emphasizing its ease-of-use and attractive look as a drawing card. The service is now available to the 15 million Macintosh users, but won't be ready for the IBM-compatible world until early next year. (*Tampa Tribune* 6/20/94 B&F14)

Microsoft Ships Test Release of Chicago. Microsoft is shipping a test version of its Windows upgrade, code named Chicago, to more than 20,000 customers, software developers and computer makers. Its general release is expected by the end of the year. "Our goal . . . is assuring that Windows `Chicago' will be a `no-brainer' upgrade," says a Microsoft VP. (*Wall Street Journal* 6/21/94 B9)

NeXT Alternative. NeXT Inc. has lined up Digital Equipment, in addition to Hewlett-Packard and Sun Microsystems, to package NeXT software with their hardware products. The new marketing strategy will help NeXT "emerge as an alternative to Microsoft," says CEO Steve Jobs. (*Wall Street Journal* 6/21/94 B9)

Will Internet Be Paradise Lost? Author James Fallows predicts that as the Internet expands, "something will have to give: either the government will stop paying, or politicians will notice that the government is paying and will impose controls, like those imposed by school boards on textbook content or by the FCC on radio and TV broadcasts. The Internet's low-visibility era of subsidized innocence will end, and the network will become as complicated as anything else." (*Atlantic Monthly*, July 94, p.34)

Computer Future Is Flat. The director of Xerox Palo Alto Research Center's electronics and imaging lab, Malcolm Thompson, says that the lab has developed a flat panel screen which is a computer and which has the resolution of a piece of paper. "If the United States wants to survive in the computer business, it better take an intense interest in this screen business, because that will the computer of the future." (*McNeil-Lehrer Report*, PBS TV, 6/21/94)

Microsoft and Stac Shake Hands. Microsoft and Stac Electronics have settled their dispute over patent infringement. Microsoft will pay Stac $1 million a month for the next 43 months for copying its data compression technology, and the two companies will cross-license all of their disk compression patents over the next five years. (*Investor's Business Daily* 6/22/94 A13)

PSI Tells Law Firm to"Cease And Desist" Advertising. Performance Systems International Inc., which provides Internet access for Arizona law firm Canter & Siegel, has ordered them to "cease and desist" their unsolicited advertising on some 1,000 bulletin boards. The law firm has raked in about $100,000 in business since their first ad was posted in April, but the husband-and-wife principals also have been deluged with obscene phone calls and "carloads" of magazines to which they never subscribed. "This is a down and dirty bunch of irresponsible miscreants," says Ms. Siegel. (*Wall Street Journal* 6/22/94 B5)

Sprint Rolls Out Wireless Services. Sprint offers two new services for professionals on the go, providing wireless connections to SprintNet for laptop or handheld computers. The first uses a nationwide 800 number for travelers with cellular modems and the second offers access through the networks of RAM Mobile Data, a joint venture between BellSouth and RAM Broadcasting Corp. (*Investor's Business Daily* 6/23/94 A19)

Too Much of a Good Thing- E-mail Overload. CEOs and other Very Important People are suffering from e-mail overload, and some are requesting to be contacted by more conventional methods, because they no longer attempt to handle their e-mail backlog. "E-mail is a powerful tool to promote communication and flatten hierarchies. But what nobody wants to admit is that people in an organization have different amounts of power and status. And that those who are better off want to restore a degree of isolation," says Rensselaer Polytechnic Institute's Langdon Winner. And it's not just the top dogs that are overwhelmed – the chairman of Computer Associates shuts down the company's e-mail system five hours a day so that everyone can get their real work done. (*Wall Street Journal* 6/22/94 A1)

Sing-along on Demand. Sega is planning an interactive karaoke system that will allow consumers to order sing-along karaoke tunes, with lyrics, over the net. The system will use an Hitachi 32-bit RISC processor and use an operating system from Integrated Systems. (*New York Times* 6/22/94 C5) [*What will they think of next?. . .Ed*]

Whirlwind Courtships Consolidate Internet Providers. Many of the nearly 100 small start-up firms that provide commercial access to the Internet are being eyed by larger communications companies as marital prospects, in the ongoing mergermania among the telecommunications, media and computer industries. The process benefits both partners – small providers get an influx of much-appreciated cash, and larger firms get an instant share of markets that are sometimes hard to reach. Says one pragmatic provider, "I think many of the smaller Internet providers hope that they'll at least get bought out and not get squashed out." (*Wall Street Journal* 6/24/94 B2)

The Big Squeeze on Dram Chips. Supplies of DRAMs (dynamic random access memory chips) are down, and PC makers are vying for what's left, with smaller companies often bidding higher prices than competitors to lock in future sales. The squeeze stems from static production on the part of Japanese memory chip-makers combined with a growing demand for more memory in the PC market. It's predicted that demand for four-megabit chips will exceed supply by 12% in the third quarter and 31% in the fourth quarter this year. (*Wall Street Journal* 6/24/94 B5)

Motorola's Mini-phone. At 3.9 ounces, Motorola's Micro-TAC Elite cellular telephone weighs less than a D-cell battery. It also can be equipped with a chip that acts as a digital answering machine, and can store up to 75 seconds of messages. (*Tampa Tribune* 6/25/94 B&F7)

3 Rs + Computer Literacy. New Brunswick plans to make computer literacy a core subject required for high school graduation, in hopes of reducing unemployment. (*Toronto Globe & Mail* 6/24/94 B1)

Bellsouth Tests Interactive TV. BellSouth will implement interactive TV in the 12,000-family metro Atlanta test area in 1995. Participants will have access to 300 specialty TV channels and 60 conventional cable TV channels, and will be able to shop, bank, play games, download movies, and enroll in courses. (*Atlanta Journal-Constitution* 6/28/94 E1)

Green Machines Are Slow to Catch on. Following last year's hoopla over the EPA's "Energy Star" program (which encouraged computer manufacturers to conform to new government guidelines on energy efficiency) demand for the "green" machines is still sluggish. Despite annual energy savings of $50 to $100 per machine which could add up to billions nationwide, most companies are driven by performance, speed and the bottom line when computer shopping. Noting that there's no price or performance differential between most green and non-green machines, one industry analysts says, "Once retail buyers become aware of the environmental benefits that are available at no added cost, the market should take off." (*Investor's Business Daily* 6/27/94 A4)

AAUP President Wants Status Quo On Internet Pricing. The new president of the American Association of University Professors has drafted a resolution in favor of maintaining flat-rate pricing on the Internet, rejecting some government moves toward sliding scale charges based on use. The measure was unanimously approved by the membership this month. (*Chronicle of Higher Education* 6/29/94 A17)

Digital Cable's Price Tag. While we all wait for the fabled 500 channels, it's useful to remember that it costs $75,000 for equipment needed to digitize and compress just one program, meaning an outlay of $450,000 to squeeze six programs into one existing channel. Smaller cable operators with only a few thousand customers are reluctant to spend $3 million for equipment to launch 40 channels of digitized premium fare, says a Cox Cable Communications vice president. (*Tampa Tribune* 6/27/94 B&F7)

Pulling the Plug tn PCs. Laptops usually come with some sort of power management built in – the power to the disk drive and screen is cut off if the keyboard hasn't been touched for a few minutes. PicoPower Technology has developed a chip that monitors what the "brain" chips inside desktop PCs are doing, cutting the power when it senses an operation that requires any period of waiting, and restoring it a split second before it's time to resume the operation. PicoPower's president says his approach reduces power consumption by an average of 70%, and keeps your machine running cool. (*Business Week* 7/4/94 p.93)

Managers Are Barrier to Telecommuting. A Conference Board survey of 155 companies shows that while more than 70% offered telecommuting options to their employees, fewer than 1% took part, citing mistrust by managers that telecommuters could be as productive while unsupervised as they are when supervised. (*Investor's Business Daily* 6/28/94 A4)

Copyright Issues Scare Multimedia Developers. Multimedia developers in higher education say their projects are being held back by difficulties in getting permission to use video clips, audio recordings, photos, and text. Some believe the colleges should exercise their rights under the fair-use provision of the copyright law, but many say colleges are too afraid of getting sued. (*Chronicle of Higher Education* 6/29/94 A17)

IBM To Put Solaris on Powerpc. IBM will offer a version of Sun's Solaris operating system on IBM PowerPC-based personal computers available next year. The PowerPC chip, developed jointly by Apple, IBM, and Motorola but currently used only by Apple, is about as fast as Intel's Pentium chip but less expensive. (*New York Times* 6/30/94 C4)

Intel Says PC Growth Driven by Communications. Intel CEO says that Intel expects PC sales to reach 100 million units a year by the late 1990s, as the result of the interdependence of PC and communications technologies. "Communications is going to shape what is happening to the PC world. Conversely, PCs are going to shape what is happening in the communications world." (*Wall Street Journal* 6/29/94 B6) [*Ed Note: 100M/year means one annually for every household in the USA. Of course, this is a worldwide prediction.*]

Cybertalk. When Vice President Gore participated last January in an on-line conference held over CompuServe, 900 people "showed up," 300 of them got the moderator's attention, and 10 of them got to match their typing skills against the Vice President's. We think he won; he's pretty fast. (*U.S. News & World Report* 7/4/94 p.70)

Sun Forms Interactive Subsidiary. Sun is forming a new business unit called Sun Interactive to build products for the phone, cable and interactive markets. It will make connections between Sun's server products, ATM networks, and set-top boxes used for interactive TV. (*Investor's Business Daily* 6/30/94 A5)

"Boss" Keys. Many applications programs now have games built into them for workers to play when the boss isn't looking – along with "boss" keys which can instantly throw onto the screen a spreadsheet or some other serious-looking display. The Gartner Group calculates that businesses lose 26 million hours of employee time (or $750 million a year) from game playing. (*Atlanta Journal-Constitution* 7/3/94 R8)

Does Telecom Growth Mean Jobs? New telecom businesses are flourishing in the metropolitan New York area as converging technologies and loosening regulations combine to encourage entrepreneurial efforts. However, economists are skeptical that the new small-scale ventures can generate enough jobs to replace those lost by downsizing in by the telephone industry. (*New York Times* 7/5/94 A12)

Flash Chips with More Flash Planned. NEC and SunDisk are developing flash memory chips that are 16 times larger than the largest flash chips currently available. Planned for release in 1997, the 256-megabit chips would could be used to substitute for miniature hard drives in portable PCs. (*Atlanta Journal-Constitution* 7/5/94 D3)

Academia Meditates on Distance Learning. The rapid expansion of technology-based distance learning in higher education has universities asking tough questions about how faculty should be compensated for teaching hundreds of students at different sites, how colleges can compete with others in different states, and how institutions can insure that courses taught at a distance are high quality. (*Chronicle of Higher Education* 7/6/94)

Newton Strategy. Apple is hoping to recover from disappointing sales of the Newton MessagePad by repositioning it as a practical device for mobile business professionals, by providing software development tools to make it easier for companies to create customized software, and by keeping a top price of $599 – half the price of most handheld computers now used by field workers. (*Business Week* 7/11/94 p.41)

Color Printing Press For Short Runs. An Israeli company called Indigo has developed a printing press that can produce color documents in short runs at fast speeds; its founder claims that even larger-circulation magazines will be able to customize their printing economically to give each individual reader a unique collection of features and ads. (*Business Week* 7/11/94 p.143)

FBI Hunt for Hacker. Kevin Mitnick is wanted by the FBI for suspicion of software and data theft from leading telecom manufacturers and service providers. Among his victims have been MCI and Digital Equipment. An ex-convict, Mitnick was described by one judge as having an "addiction problem" with computers, similar to a drug or gambling addiction. During a six-month treatment program he was prohibited from touching a computer or a modem, but the treatment seems to have failed, and one detective says: "I've always considered him dangerous. I had to go underground. If he targets you, he can make your life miserable." (*New York Times* 7/4/94 A1)

TV Chat. America Online and Capital Cities/ABC will offer AOL subscribers the opportunity to tap into an online newswire, a celebrity "chat", and interactive games from ABC Sports. ABC is the last of the Big Four broadcasters to wade into Cyberspace. (*Wall Street Journal* 7/7/94 B5)

MIT and Cern to Develop Internet Standards. MIT and the Center for European Nuclear Research will collaborate on developing international standards based on World Wide Web software architecture, ensuring compatibility between future Internet navigational systems. The alliance, to include

companies from the U.S. and Europe, will also address security and privacy issues in electronic data transmission. (*Wall Street Journal* 7/8/94 B3)

Phone Numbers for Life. AT&T's True Connections plan offers you a phone number for life (at least as long as you pay your bill). The new "500" area code numbers will enable subscribers to program their existing phone numbers to follow them wherever they go. The numbers could ring in sequence – e.g., at the office, then at a cellular phone, then at home. Pending FCC approval, the new service will be available in September. (*Washington Post* 7/8/94 F1)

Are Your Documents Full of Glyphs?. A Xerox technology known as glyphs will allow documents to carry thousands of characters of information placed unobtrusively in gray background patterns. One possible use: "If you see a spreadsheet in an annual report, it sits there, lifeless on the paper. But if there was a glyph border that had the mathematical model of the spreadsheet, you could scan that into a computer and make it come to life." Another possible use would be to encode info about the recipient of a direct mail piece or a survey, for ease of processing when the document is returned. (*New York Times* 7/10/94 Sec.3, p.9)

CD Licensing Activities Under Scrutiny. The Justice Department is investigating whether Sony Corp. and Phillips Electronics NV violated antitrust statutes through a practice called patent pooling. The investigation focuses on how the companies jointly charge patent-licensing fees to make and market compact disks. After cross-licensing their patents in the 1970s, the two firms have collected millions of dollars a year in fees, while ensuring that new developments in CD technology remain compatible with their products. "This whole story is a classic example of how technology that originally started in the U.S. is now completely in the hands of foreign companies that have basically used the patent system in a way it wasn't intended to be used," says the executive vice president of the largest independent U.S. disk maker. (*Investor's Business Daily* 7/12/94 A1) [*Ed: How was the patent system supposed to be used?*]

U.S. Governement to Pay Royalties on Clipper Technology. The National Institute of Standards and Technology has agreed to pay an MIT computer scientist royalties for two patents related to Clipper-chip technology. NIST previously had disputed the patent claims, but then decided that licensing the technology was the best way to avoid future legal concerns. (*Wall Street Journal* 7/12/94 B6)

Dell's Retail Sales Experiment Fizzles. Citing financial losses, Dell has stopped shipping PCs to its five retail partners – CompUSA Inc., Best Buy Co., Sam's Club, Price CostCo, and PC World in Britain – and will concentrate on its dynamic direct sales strategy. However, the company will continue to explore new methods of reaching consumers, such as delivering catalogs of products and services electronically through online services and the Internet, and establishing kiosks in malls and airports. (*Investor's Business Daily* 7/12/94 A3)

Software for Just-in-time Learning. Northwestern University's Institute for Learning Sciences, with funding from the Pentagon and Andersen Consulting, does leading-edge research into how people learn, and applies it to developing multimedia software aimed at creating an electronic, just-in-time teacher. The ILS's ASK system responds to student queries with special "case retrieval" software that gives students access to a video database of subject experts who relay "stories" to answer the questions. "ILS Director Roger Schank and the ILS are 'pioneers on a wild frontier'" in teaching and learning, says the NSF's John Clement. (*Business Week* 7/18/94 p.74)

A Shakeout for On-Line Services? Some analysts are predicting a glut of information services on the net, and the v.p. and general manager of Delphi says, "There's a shake-out on the horizon. The numbers are growing rapidly, but they won't be high enough to accommodate all the companies who are coming into the market." The 10 largest revenue producers among the online services brought in approximately $500 million last year. (*New York Times* 7/12/94 C1)

Thomson And Sun Microsystems Form New Company. Thomson Consumer Electronics and Sun Microsystems will form a company to sell equipment to enhance the networks and services of telephone and cable companies. The alliance will have the advantage of being able to offer a full range of products, from video servers to set-top boxes. Thomson manufactures TVs and other appliances under the GE and RCA labels, and Sun's relationships with cable and phone companies earn it more than $1 billion a year in revenues. (*Miami Herald* 7/12/94 C3)

Electronic Nose. A British company has developed the first "electronic nose," capable of measuring and recording smells digitally. AromaScan Plc says its invention will revolutionize aroma analysis in the food, drink and perfume industries. The sniffers currently go for $38,570 apiece. (*USA Today* 12/11/94 A1)

Culture Clash on the Internet. In her new book, "Who Owns Information?", information technology lawyer Anne Wells Branscomb says it will be hard to resolve the "culture clash between the academic community, which is used to general access, and the business community, which is paying a per-use basis, per piece of information." (*Newsweek* 7/18/94 p.54)

Yellow Pages for the Web. An MIT graduate student has developed the closest thing so far to Yellow Pages of commercial activities on the World-Wide Web. Commercial Services on the Net *(http://tns-www.lcs.mit.edu/commerce.html)* currently consists of an alphabetic list of 60-80% of the existing commercial Web sites. The developer plans to add a search capability and a category breakdown. (*Internet Business Report* 7/94 p.3)

CNN Wants "talkback" from the Net. A new one-hour CNN program hopes "to use technology to bring us closer together. The cable station is making deals with MCI and CompuServe that will allow two-way communication with viewers, and the MCI-CNN link will provide teleconferencing based on compressed video sent by phone lines. (*Atlanta Journal-Constitution* 7/11/94 A1)

Finger Triggers Privacy Alarms. Many college and university computer system administrators are responding to rising concerns over misuse of the Finger tool with modifications that restrict the information users can glean, and some have eliminated it altogether. Critics note the tool violates privacy – it provides information about where people are logging on and when they're doing it – and security – crackers can use it to obtain information that can help them break into computer accounts. "A telephone directory is a great thing, until you realize that people who don't have your best interests in mind can use the information in it to do terrible things to you," says one university computer system administrator. (*Chronicle of Higher Education* 7/13/94 A15)

Beware of Creeps in Cyberspace. "Online computer services are becoming the pedophile's playground of the '90s," says a Florida Department of Law Enforcement official, and the FDLE and U.S. Customs agents have come up with a list of warning signs for parents: Your child spends an inordinate amount of time (more than a couple of hours a day) online; the screen goes blank when you walk into the room; your child is using a lot of diskettes to retrieve material; you find diskettes hidden. Also: watch for computer files that end in -GIF, -GL, -TIF, or -JPG, which are likely to be picture files. (*Miami Herald* 7/14/94 A1)

IBM Goes All Out in Disk Drive Capacity. Researchers at IBM have outdone themselves, exceeding their own goals for giant magnetoresistance (GMR) technology. Commercial versions of GMR heads will sense 10 billion magnetic dots per square inch, resulting in thumbtack-size computer disk drives with 100-megabyte capacities by the end of the decade. (*Business Week* 7/11/94 p.145)

Microsoft Settles Antitrust Inquiry. After four and a half years, Microsoft has agreed to end allegedly anti-competitive licensing practices for MS-DOS and Windows operating system software. The European Commission, which also had been investigating the company's business practices in Europe, signed a similar accord Friday. Industry experts were split on how the agreement would affect future prices for Microsoft products. (*St. Petersburg Times* 7/17/94 A4)

World Book and Britannica Stick to Their Guns. With dwindling sales in print encyclopedias, the two industry leaders are re-evaluating whether to join the CD-ROM revolution and compete directly with $99 products from Grolier, Compton's and Microsoft. "We think of World Book as the Cadillac of encyclopedias. We're not going to sell it like a Yugo," says the president of World Book's sales subsidiary. Nevertheless, both companies are now offering CD-ROM versions of their products as a bonus for buyers of the print set. Although either CD-ROM may be purchased alone – World Book's for $395 and Britannica's for $1,595 – the companies are not pushing the discs as a replacement for the print product. (*St. Petersburg Times* 7/17/94 H8)

Freenets Raise the Ire of Commercial Providers. State-supported projects to provide citizens with Internet access for little or no money are upsetting commercial service providers who want to sell that same access for $20 or so a month. A NYNEX official noted the company "does not oppose the use of public libraries and other facilities to disseminate access to the Internet." But when a university hooks up a FreeNet and provides access to commercial entities, "we think that's bad public policy and a waste of taxpayer funds." (*Chronicle of Higher Education* 7/27/94 A19)

Rubbernecking on the Internet. Traffic was brought to a virtual standstill on the Internet July 19, caused by people looking for the latest pictures of the Jupiter-comet crash. (*Information Week* 8/1/94 p.8)

1984 All Over Again. Philip R. Zimmerman, who is the author of the data encryption software called Pretty Good Privacy (and who is the subject of a criminal investigation because his software has appeared overseas, in apparent violation of U.S. export-control laws), says: "It's possible for Government computers to automatically scan for keywords in our electronic mail. This has a bad effect on democracy; it's like '1984'." (*New York Times* 7/25/94 A6)

Movies on CD-ROM. Image Entertainment will begin this fall distributing full-length movies on CD-ROM for $10 to $20 a title. The movies will be marketed through discount, video, music and computer stores, and will be available in both IBM and Macintosh formats. Each title will include ancillary materials, including electronic press kits and interviews with directors and performers. "This is going to be a very limited market, mostly because people won't be willing to view movies on a 14-inch monitor for two hours," says one industry skeptic. (*Business Week* 8/1/94 p.68, *Wall Street Journal* 7/25/94 B7)

Microsoft's Good News, Bad News. When you're on top of the world, the only way to go is down, and Microsoft's officers are especially cautious about next year's prospects, particularly contrasted against the stunning successes achieved in the fiscal year just ended. "We are really worried about saturation," says an executive V.P. In addition to lower PC sales, Microsoft fears irrational price cutting by "desperate competitors" and widespread piracy of Microsoft software. Despite the hand-wringing, the company plans to spend an extra $100 million on marketing and will increase its worldwide work force by 17%. (*Wall Street Journal* 7/25/94 B2)

IBM Abandons Ambra. IBM is shutting down its Ambra subsidiary, which manufactured IBM clones to compete against low-priced IBM competitors such as Dell and Gateway 2000. IBM intends to emphasize its own brand name in future personal computers. (*New York Times* 7/29/94 C3)

Internet Statistics. *Internet Info* reports that as of July 15, there were more than 17,000 company domains registered with the Internet. Predictably the companies with the "heaviest" presence (defined as 25 or more networks) were primarily defense contractors and telecommunications firms. California had the largest concentration of .com activity. (*info@internetinfo.com*)

Computer Pornography Suit. The conviction this week of a couple charged with transmitting obscene photos over their computer bulletin was the first such obscenity trial that was held in the locale where the material was received (Memphis, Tennessee) rather than where it was sent (Milpitas, California). The defendants charged that the federal prosecutors chose a Bible Belt city where they could count on a sympathetic jury. (*Atlanta Journal-Constitution* 7/29/94 A3)

McNealy Decries Microsoft Monopoly. In a *Wall Street Journal* op-ed piece, Sun Microsystems' CEO Scott McNealy calls for open standards in computer operating systems, saying Microsoft's monopoly in the DOS/Windows market is unhealthy: "This means one company has a virtual lock on a language that is now as critical to the world economy as the written and spoken language." (*Wall Street Journal* 7/27/94 A10)

Snail Mail Update. The Dutch post office is taking the lead in providing mail service beyond its borders, and soon will be delivering all of China's and most of Canada's European mail. The trend is spreading, fueled by competition from private delivery companies and technological rivals such as fax machines and e-mail. "The barriers will disappear, and people will make up their own minds about how they're going to ship their mail," says the president of the Dutch PTT Post. (*Wall Street Journal* 7/28/94 B1)

Bypassing the Music Industry. Two seniors at the University of California – Santa Cruz offer would-be rock stars a way to get their music distributed electronically to millions without ever signing a record contract through their Internet Underground Music Archive. (*Details* 7/94 p.118)

Students Envision Computers of the Future. Apple Computer's third annual design competition drew entries such as a computer shaped like a hand-held mirror for hospital patients to use to exchange messages with physicians and seek information from online medical databases; a shoulder-mounted computer including a camera, microphone and speakers, enabling a remote viewer to take a virtual "walking tour" of another country; and a system called "Galen" designed to enable nurses to connect to medical databases and read the results in 26 languages. (*Chronicle of Higher Education* 8/3/94 A15)

Fighting Music Bandits on the Information Superhighway. The Recording Industry Association of America is lobbying Congress to approve a copyright law that would provide royalties to recording artists and record companies for music that is digitally transmitted. Current copyright law provides royalties only to songwriters and publishers.

Chip Update. Texas Instruments will join with Hitachi Ltd. to build a $500 million dynamic random access memory chip factory in the U.S. Analysts expect the $18 billion worldwide Dram chip market to expand rapidly as advanced computers and telecommunications products require more memory to function. (*New York Times* 8/2/94 C3)

Good News/Bad News on the I-way. The good news is, a survey of 1,000 people conducted by the Consumer Technology Group surprisingly shows that nearly half of American households are already cruising the information superhighway. The bad news is that 40% think they're going the wrong way. (*Tampa Tribune* 8/1/94 B&F 10)

Digital Targets Business on the Net. Digital Equipment Corp. has created a new unit, the Internet Business Group. The unit will sell products geared toward providing security against cyberprowlers and navigating the Internet. DEC also will use the Internet for marketing its other products. Users will be able to remotely "test drive" the company's advanced Alpha processor and request other information on DEC products. (*Investor's Business Daily* 8/3/94 A12)

CD-ROM Sales Up, Use Down. Multimedia players for PCs continue to sell at record levels, but once home, they often end up gathering dust in a corner. A Dataquest survey shows that 40% of the people questioned said they never used their CD-ROM players, and 54% do not plan to buy additional software. (*Wall Street Journal* 8/4/94 B6)

Money's No Object. Only 42% of information service managers surveyed by Computer Intelligence Infocorp say cost is an important factor when purchasing computer network servers. What is important are reliability (80%) and performance (76%). (*Investor's Business Daily* 8/4/94 A3)

Universal Access Compared to What?. While the debate continues over what universal access to the information superhighway actually means, Mitch Kapor reminds us, "Meanwhile, 98% of U.S. households have TVs (only 93% have telephones) and all those people in the 98% paid for their TVs – television is important enough that people go out and actually spend money on it." (*Technology Review* August/September '94 p.42)

More I-way Survey Results. According to a new survey of 1,000 people conducted by Porter, 33% say they're "going the speed limit in the right lane" on the information superhighway and 18% say they're "on the entrance ramp." A speedy 11% boast they're "passing everyone on the left." "There's a gap in perception in terms of where business folks think consumers are and where consumers really are," says a public relations firm spokesman. (*St. Petersburg Times* 8/3/94 E1) [*Ed: How far can an analogy be taken? Time will tell.*]

IBM Overhauls UNIX. IBM has overhauled its UNIX operating system, and plans to ship the updated version August 12. The new software contains a kernel robust enough to support future IBM symmetric multiprocessors, as well as IBM's Workplace technologies, an initiative allowing many core functions to be shared among IBM's various operating systems. (*InformationWeek* 8/8/94 p.16) [*Ed: ``Its" Unix operating system?*]

Happy Birthday to the Internet. Twenty-five years ago the Internet began with the creation of ARPANet, funded by the Department of Defense's Advanced Research Project Agency. Vint Cerf, president of the Internet Society and one of the people who participated in that ARPA project, says: "You don't know how far you've come until you stop and look back." (*Newsweek* 7/8/94 p.56)

Software Patents. The Court of Appeals for the Federal Circuit has ruled that a software program can be patented because, by setting switches in a computer's processor, it creates a new circuit (and thus a new machine). Patent lawyer Donald Chisum says that "the court has broadly affirmed the patentability of software – as long as it is operating a machine or causing a physical result." (*New York Times* 8/8/94 C2)

Computer Viruses Are A "Life Form". British physicist Stephen Hawking says a computer virus fits the definition of a living system even though it has no metabolism of its own, and instead uses the metabolism of a host computer for its parasitic existence. "I think computer viruses should count as life. I think it says something about human nature that the only form of life we have created so far is purely destructive. We've created life in our own image," says Hawking. (*St. Petersburg Times* 8/8/94 p.8)

PowerPC Still Waiting for Takeoff. Although the product was delivered right on time, IBM, Motorola and Apple are still waiting for the big payoff from their PowerPC joint venture. Unfortunately, it's a chicken-and-egg problem, with consumers waiting until there are plenty of hardware and applications to go along with it, and developers sitting on the fence waiting for market momentum before adopting the new technology. One problem is that Apple and IBM still lack a common hardware standard, using different methods to deal with peripherals such as keyboards and floppy drives. (*Business Week* 8/15/94 p.96)

Internet Keeps on Growing and Growing. The Internet Society says there are now 3.2 million reachable machines on the Internet, and 1 million new hosts were added during the first six months of 1994, with much of the growth attributable to growth outside the world in more than 80 countries. For more info: *http://info.isoc.org*. (ISOC Release 8/4/94)

Internet Access Providers. The Maloff Company estimates the following ranking of Internet access providers, with approximate percentage of the total U.S. IP marketplace, based on revenue, as of March 1994: PSI, 13%; UUNet/Alternet, 12%; Sprint, 12%; IP Resellers, 10%; Nonspecified Regional Nets, 10%; ANS, 9%; NETCOM, 7%; CERFnet, 4%; Colorado Supernet, 4%; NEARnet, 3%; World, 3%. (*Internet Business Journal,* July-August 94, p.8)

Counting Computers. Although many estimates place the number of machines on the Internet at 20 to 30 million, some Internet demographers say that only about two million persons are actually reachable over the net. They question the assumptions on which some of the estimates are made (such as the assumption that each machine represents about 10 people), and they suggest that many people in organizations are walled off by security measures from much of the potential incoming network traffic. (*New York Times* 7/10/94 A1)

Techno-competents Are the New Worker Elite. One out of every four new jobs goes to a technical worker, says the Bureau of Labor Statistics, which predicts that technical jobs will represent a fifth of total employment within a decade. (*Fortune* 8/22/94 p.56)

CIA Plans One-way Mirror on the Internet. The CIA plans to start using the Internet for gathering information, but will configure its systems to prevent file transfers in the opposite direction, because the agency is "keenly aware" that by connecting to the net it increases the danger of security breaches by hackers. (*Atlanta Journal-Constitution* 8/11/94 D2)

You Either Know the Feeling, or You Don't. A survey by a Teaneck, N.J., research firm reports that four out of ten computer users have felt like throwing their PCs out of the window. The survey provides no information about what the other six people feel like doing. (*Wall Street Journal* 8/11/94 A1)

CERT Redefines its Mission. The Computer Emergency Response Team has been overwhelmed by requests for help in recent months and has decided to concentrate its efforts on battling major threats to the global Internet. Internet service providers will be responsible for resolving run-of-the-mill security incidents. CERT plans to step in immediately if the problem seems serious. "If you have a headache, start with taking an aspirin. If you have a headache and a fever, we'll want to take a look at that," says the manager of CERT operations. (*Chronicle of Higher Education* 8/17/94 A24)

Thinking Machines to File for Chapter 11. The Thinking Machines Corporation, the Cambridge, MA, supercomputer company that pioneered massively parallel computers linking hundreds or thousands of microprocessors in a single machine, is declaring Chapter 11 bankruptcy, after a losing battle with larger companies such as IBM, Cray Research, and Intel over steadily decreasing Defense Department contracts. The company is hoping to sell pieces of its technology in order to recoup money to repay $125 million from institutional investors. "Sadly, the parts seem to be worth more than the whole at this point," says TM's CEO Richard P. Fishman. (*New York Times* 8/16/94 C2)

Data-Mining Is the Next Big Thing for Supercomputers. Big credit card companies, banks, airlines and insurers have discovered massively parallel processing in an effort to divine which consumers are likely to buy what products and when. A Gartner Group VP predicts sales of parallel systems could expand tenfold to $5 billion by 1998 as a result of this new application. While marketing folks are waxing euphoric, one business professor warns the fallout could be nasty if companies start abusing their newfound info: "The companies doing this have a big responsibility. Otherwise there will be an information Chernobyl." (*Wall Street Journal* 8/16/94 B1)\

Online Services Have Data Mines, Too. The online service you use has been compiling data on you too, including your social security number, credit card number, demography and interest areas. Using this and other data, CompuServe offers a service called CompuTrace, which offers the last known address for any person in the U.S. A similar service will tell you how long someone has had a particular phone number or lived at a particular address and who else lives there, and yet another service provides information on how to obtain driving records, state by state. A bill was passed by the House last month that would require all telecommunications companies, including online services, to tell consumers what information is being collected, how it's being used, and provide an opportunity to opt out. (*Tampa Tribune* 8/15/94 B&F 3)

Telecom Companies Balk At Cost Of Wiretap Technology. The Digital Telephony bill, introduced in Congress last week, would require that telecommunications companies modify their networks so that law enforcement agencies can continue to conduct wiretaps and trace messages as technology evolves. The bill includes $500 million to reimburse companies for their trouble, but the Electronic Frontier Foundation estimates that it will cost five to ten times as much. The companies now want language included guaranteeing full reimbursement for complying with the law. (*BNA Daily Report for Executives* 8/12/94 A14)

Telecommuting Spurs Demand For Isdn. Integrated services digital network is finally coming into its own, as telecommuters snap up the high-speed data connections necessary to access many corporate networks. Dataquest estimates the number of ISDN lines using a basic rate interface will reach 226.4 million by the end of this year, almost double from 1993. That number will hit 335 million lines by the end of 1995. (*Investor's Business Daily* 8/15/94 A4)

International Freenet Conference. Freenet activists from around the world are meeting to work on building community networks into a major political force demanding free and open access on the info-highway. As the Internet becomes increasingly commercialized, freenets are likely to gain strength in their battle to maintain a public, non-commercial access lane for traffic. (*Ottawa Citizen* 8/16/94 D1)

Japan Wants Private Sector To Build Network. Japan's Ministry of International Trade and Industry wants the private sector to take the lead in building a $750-billion national fiber-optic network that would link all Japanese homes and businesses by 2010. MITI's position is an about-face from its usual pro-active stance on government/industry technology projects, and U.S. companies are eager to play a major role in constructing the Japanese Infobahn. (*Wall Street Journal* 8/15/94 A1)

Latin America Is PC Boomtown. Sales of personal computers in Latin America totaled $2.4 billion in 1993, and are expected to more than double over the next five years. Brazil was the largest national market last year, increasing more than 34% over the previous year. Sales in Argentina and Chile grew by 40% last year, while Mexico's sales declined by 5% due to the uncertainties of NAFTA and pre-election jitters. Nevertheless, Mexican sales accounted for 30% of all sales in the region. (*Miami Herald* 8/15/94 p.14)

An Update on Standards Relevant to USENIX Members

by Nick Stoughton
USENIX Standards Report Editor
<nick@usenix.org>

The following reports are published in this column:

- POSIX.0: Guide to POSIX OSE
- POSIX Conformance Testing
- POSIX Interpretations
- SPEC 1170: A Developer's Reply

Our Standards Report Editor, **Nick Stoughton**, welcomes dialogue between this column and you, the readers. Please send your comments to *<nick@usenix.org>*.

Report on POSIX.0: Guide to POSIX OSE

Kevin Lewis <klewis@gucci.enet.dec.com> reports on the July 11-15, 1994 meeting in Nashua, NH.:

Hello to you who have been following with great vigor the progress of the POSIX.0 work. I was absent from the April POSIX meeting, helping my wife have our fifth baby. It was a girl, and she is a party animal so the July meeting afforded me some rest.

First, let me bring you up to date on the status of the group's work. At the conclusion of the July meeting, the voting status on the guide document (draft 16) was at 89% affirmative. [NOTE: some of you may have seen a draft 16.1 floating around. This is a draft that was created just after the draft 16 recirculation in IEEE started, in order to address some concerns at the international level, i.e., JTC1/SC22/WG15]. This percentage is based on 45 affirmative votes which were received at the close of the recirculation ballot and 10 affirmative votes which had initially been negative but were converted to affirmative during the ballot resolution process. Seven votes remain negative. It is highly possible that with the probability of some last minute resolutions that will be taking place with the weeks following the July meeting, the affirmative vote could reach 91%.

Draft 16.1 was submitted to WG15 for an SC22 letter ballot which will conclude on August 5. The comments arising from this letter ballot will be reviewed and dispositions created by the POSIX.0 technical reviewers. The resulting document changes will be folded into the recirculation resolutions for production of draft 17. This version will be submitted in late August/early September for another 30-day recirculation starting October 1. Parallel to this, the working group will prepare draft 17 for submission to the IEEE Review Committee (RevCom) and also submit it for subsequent SC22 DTR ballot.

I anticipate that at the conclusion of this recirculation, the document will be ready for submission to the IEEE Standards Board for approval.

There are approximately 19 unresolved objections remaining. These fall into the following areas: Layered APIs, External Environment Interfaces within the Reference Model, and the inclusion of Public Specifications.

I do believe that we are now reaching the end of the road on this work. For those who are interested, we are planning to discuss RevCom preparation along with reviewing any comments that have come in from the second recirculation. It very well may be that the October meeting will be our last meeting, unless some new work comes our way. But please don't ask me about that right now. I only want to get this done so I can save some trees.

Report on POSIX Conformance Testing

Kathy Liburdy <kliburdy@eng.clemson.edu> reports on the July 11-15, 1994 meeting in Nashua, NH:

Conformance testing in the IEEE Portable Applications Standards Committee (PASC) standardization effort is a composite of several bodies each of which contributes to the overall goal of enhancing the quality of the POSIX standards environment. This report summarizes the state of conformance testing at the time of the POSIX meeting in July, 1994.

The Steering Committee on Conformance and Testing (SCCT) discussed issues related to keeping test methods and the corresponding base standard in synchronization. Five cases were identified which could result in discrepancies between the test methods and base standard. If there is a discrepancy because there is an error in the approved test method standard, because existing base standards specifications are changed, or because an official interpretation of the base standard results in the identification of a problem in the TM standard, then the proposed solution is to initiate an amendment for the TM standard. It is less clear what action should be recommended if the base standard must be revised, either due to identification of an error in the approved base standard or due to an interpretation of the base standard which requires modifying the base standard.

The issue which emerges from these latter two cases is one of timing: Should the TM reflect the existing base standard even though modifications are imminent, or should the TM reflect the changes which are assumed to be applied to the base standard at some time in the future? While acknowledging that there is no clear answer to this question, the SCCT recommended that PASC adopt the policy of requiring the test methods for an approved standard to reflect the existing text of the base standard at the time of adoption of the TM standard.

Due to the resignation of Dave Hollenbeck from the SCCT, Barry Hedquist was nominated to serve the remainder of Dave's unexpired term. His nomination was approved.

Jerry Powell, a member of the SCCT and Lead Rapporteur of the WG15 Rapporteur Group on Conformance Testing (RGCT) reported that the RGCT met May 9-10 in Tokyo, Japan. Roger Martin convened the meeting as Jerry was not able to attend. The activity highlights of this meeting included the forwarding of the US contributed Harmonization Conformance Testing Terminology report to SC22/WG15. The purpose of this report is to explain the unique definitions used in the PASC Test Method standards in relation to similar terms used in the ISO/IEC testing standards. The next meeting of the RGCT is scheduled for October 24-25, 1994 at Whistler Convention Center in Whistler, British Columbia, Canada.

The PASC Test Methods working group discussed the appointment of a new chair due to the appointment of the current chair, Lowell Johnson, as PASC chair. Members discussed the possibility of a merger of the SCCT and 2003, with Roger Martin being a likely candidate for the chair of the new SCCT/2003 group.

The 2003 working group continued to work toward the final draft of the standard Rules and Guidelines for Test Method Specification for Measuring Conformance to POSIX. This standard is a revision of the existing standard for test methods development (1003.3-1991). Unresolved topics include the use of charts to explain test result codes and the introduction of a construct to permit formal test specifications in support of automated testing. A request has been issued for a ballot slot, and a consensus on these topics is expected to be achieved by the time the ballot slot becomes available.

Progress reports from groups developing test methods for base standards were given in the 2003 working group meeting. Ongoing efforts include 2003.2 (Shell and Utilities), 2003.4 (Real Time), and 2003.5 (Ada Language Binding).

The Automated Testing BOF met Wednesday afternoon, July 13. A brief report was given on the Invitational Workshop on Automated Testing hosted by NIST in cooperation with Clemson University. The goals of this workshop included reviewing existing and emerging technologies for automated specification and development of test methods, exploring the relationship between automated testing and standards development, and establishing a forum for the continuing exchange of information between experts working in this area. An overview of the workshop is scheduled to appear in the September issue of IEEE Software.

I gave an overview of the development of test methods for the Ada binding to POSIX. This work is applying the Clemson Automated Testing System (CATS) to assist in the development of test methods. The development process includes the identification of requirements in the base standard which must be tested, the development of a strategy for testing one or more requirements, and a formal specification of the test. This formal specification can be automatically translated into a test and executed on an implementation under test, thereby providing rapid feedback to the test developer which can be used to improve the quality of the test methods.

Shane McCarron reported on the progress of the Assertion Definition Language (ADL) Project. Two prototype test suites are being developed with ADL to assist in the evaluation of the ADL technology. Mt. Bonnell will be developing a test suite for TET 1.10, and Applied Testing and Technology will develop a test suite for OMG's Common Object Request Broker Architecture version 1.2. Because these efforts are part of the research project supported by X/Open, MITI and Sun

Microsystems Labs, the resulting test suites will be made available under the same copyright and grant of rights as the ADL project itself. For more information on this project, contact Shane McCarron at *ahby@naps.com.*

The July issues of the newsletter *Automated Testing in Open Systems* (*OATS*) was distributed at the POSIX meeting. This issue includes articles about the ADL classroom experiment, PHIGS Validation Tests developed by NIST, and the NIST Advanced Technology Program's initiative in Software Technology. Copies may be requested by contacting me, Kathy Liburdy at *kliburdy@eng.clemson.edu.*

Report on POSIX Interpretations

Andrew Josey <a.josey@novell.co.uk> reports on the July 11-15, 1994 meeting in Nashua, NH.:

I attended the IEEE PASC meeting in Nashua, NH on July 12th and July 13th in my role as Portable Applications Standards Committee (PASC) Vice-Chair for Interpretations (VC/Int).

This role involves coordinating interpretation requests and responses for all PASC standards, notably, although not confined to, POSIX.1 (System Interfaces) and POSIX.2 (Shell & Utilities).

I seem to have gotten this job because I complained last summer that several of my interpretations had not been processed after 18 months. In the last 9 months, we have gotten the process back on track and since January, have finalized nearly 100 requests and published four interpretation documents.

I attended two sessions, POSIX.1 on Tuesday, which spent the time reviewing interpretation requests – approximately 20 requests were covered (8 of these for 2003.1, the test methods for POSIX.1) – and POSIX.2 on Wednesday which covered nearly 30 interpretation requests.

Many of the interpretation requests against POSIX.2 highlight defects in which the standard has changed historical practice. A large number of potential corrections are being identified this way.

Interpretation requests are accepted for approved standards only. This means one of the following:

Approved IEEE PASC Standards:

System Interface Standard	IEEE Std 1003.1-1990 (ISO 9945-1:1990)
Real Time Extensions	IEEE Std 1003.1b-1993
Shell and Utilities	IEEE Std 1003.2-1992 (ISO/IEC 9945-2:1993)
Ada Bindings for POSIX	IEEE Std 1003.5-1992
Fortran 77 Bindings	IEEE Std 1003.9-1992
Test Methods (Concepts)	IEEE Std 1003.3-1991
Test Methods for 1003.1	IEEE Std 2003.1-1993
OSI API Standards	IEEE Std 1224-1993, IEEE Std 1224.1-1993 (X.400), IEEE Std 1224.2-1993 (X.500)
	1326/27/28 are related language bindings & Test methods

The most active to-date interpretation groups are 1003.1-1990 (System Interface) with 69 requests and 1003.2-1992 (Shell & Utilities) with 68 requests. The next most active is 2003.1 with 19 requests.

Most active of the last quarter is 1003.2, with 11 requests (however many of these are multi-parted including, for example, a 60-part request on vi!). POSIX.2 is the largest standard produced to-date by PASC, and now that folks are trying to achieve conformance, we do expect to see a large number of requests. In my view this is an immature standard which needs to have test methods and systems implemented before we truly get to a usable standard. It has to go through the same life cycle as POSIX.1 (i.e. the first revision will be a better product than the original).

I tend to take advantage of the working groups meeting at the PASC quarterly meetings to make them act as specialist subgroups. Actual interpretation responses can only be approved by the interpretations group mailing list (this allows full participation by all interested parties). However, in most cases the mailing list approves what the specialist group comes up with.

What can be done through the interpretation process?

Interpretation can explain and clarify the intent of the standard: it cannot make an alteration to the original standard or supply consulting information.

Changes to the standard can be made only through revision or supplement to the standard, which then go through the normal balloting process.

The PASC has guidelines for the interpretation groups which give sample pro forma responses that should be used in answering a request; for example:

- The unambiguous situation: "The standard clearly states . . . and conforming implementations must conform to this."
- The "defect" situation: "The standard states . . . and conforming implementations must conform to this. However, concerns have been raised which are being referred to the sponsor."

All in all, there are eleven pro forma responses for various situations.

A major result of the interpretation process is to highlight defects and/or ambiguities in the standards, and pass these on to the working group for the next revision. In particular, with the revision to 1003.2 (1003.2b) the work group is actively making corrections based on these interpretations. Should the next ballot fail, they intend to make further changes based on interpretation discussions at this meeting. (It's likely that the sections on `vi` will need rewriting, since 60 interpretation comments were processed).

The POSIX.1a working group is also actively monitoring POSIX.1 interpretations.

In the last quarter, PASC introduced an electronic process for submitting Interpretation requests. The latest procedure for requesting an interpretation is outlined below:

Requests for interpretations should be addressed to Secretary, IEEE Standards Board IEEE Standards Department 445 Hoes Lane P.O. Box 1331 Piscataway, NJ 08855-1331 Email: *stds.pasc-interp-request@ieee.org*.

Electronic mail is the preferred medium and flat ASCII format text, the preferred format to allow committee circulation.

Requests for interpretations should include:

- The specific designation of the standard, including the year of publication.
- The specific subsection being questioned.
- The applicable conditions for the case in question.
- A suggested correction, if applicable.

Submissions by electronic mail are processed as follows:

- The Interpretations requester sends the request to *stds.pasc-interp-request@ieee.org*.
- The IEEE acknowledges receipt of the request to the requester, with a copy of this acknowledgment to the PASC Vice Chair of Interpretations.
- When the PASC VC/Int has assigned a number to the interpretation, he or she informs the IEEE and the requester of the assigned reference number.
- After consideration by the interpretations committee, which may involve many exchanges of correspondence, the inquirer will be notified by the PASC VC/Int of the decision of the subgroup. The PASC VC/Int includes a note to the requester asking for their acknowledgment of receipt.
- Decisions will be published from time to time in cumulative form and may be ordered from the IEEE.

The IEEE PASC VC/Interpretations can be reached electronically on *stds.pasc-vc-interp@ieee.org*.

Last, but not least, I received a copy of the new POSIX 1003.1b-1993 standard at the meeting (it has a blue cover and includes POSIX.1-1990). PASC obviously knew something that I did not, since the very next day in came the first interpretation request against that standard!

SPEC 1170 – A Developer's Reply

Nicholas M. Stoughton
<nick@usenix.org>

Following my editorial in the previous issue of this newsletter, I would like to take the opportunity of allowing one of the X/Open 1170 consortium members an opportunity for a personal reply. The following is excerpted from a note to me from Mark Brown of IBM in Austin. I should point out that this is a personal reply, and not a statement from IBM!

I was a little disturbed when I read your editorial on SPEC 1170; while it is certainly one angle from which one can view the X/Open consortium, I think that I'd like to present another one.

I can tell you straight out that while this may seem to be a bonus to the author, to an applications developer it means that they are Less Than Totally Useful. I have stopped trying to count the times that my customers have asked me, after I've spent 30 minutes or so evangelizing for POSIX (I work on the POSIX.2b committee myself): "so, how portable will my code be if I am using X" where X is Shared Memory or Sockets or whatever. All those missing pieces mean that the Standards don't really deliver the goods to the people who need them the most.

1170 was started due to *customer demand.* ISVs couldn't use POSIX or XPG to get the portability they needed, due to those "missing pieces."

The end of systems programming in UNIX? Bwa-ha-ha-ha! Hardly. Did the SVID cause this? Certainly POSIX hasn't. Neither will 1170. It simply plugs a lot of holes. Heck, there were a lot of things that the group *wanted* to put in but didn't. For example, where are all the floating point interfaces?

It is really easy to put a sinister face on 1170; the truth is simply that it is a market response to a perceived gap in the published standards.

The Bookworm

by Peter H. Salus
<peter@uunet.uu.net>

Rave reviews

Let me start out on my good foot (saving the other to kick folks with, later in this column).

First of all, *!%@:* by Frey and Adams is out in a fourth edition. Back in 1989, the first edition had just under 300 pages, half of which were maps. The maps have been dropped. Nonetheless, the new edition is over 600 pages and costs a mere $9.95. This was a useful and worthwhile book a lustrum ago: it is now nearly indispensable. My gold star award to Donnalyn, Rick and ORA!

Second, there are the *4.4BSD manuals*. From USENIX and ORA. Five volumes. A CD. A CD-Companion. I keep my 4.1s, my 4.2s, and my 4.3s at my tableside. At $150 for the manuals and the CD-ROM, it's another bargain. I use my BSD docs, even though I'm running SVR4, because they're good. Use some today!

Third, there's Linux. Inspired by Andy Tanenbaum's Minix, Linux was created as a hobby project by Linus Torvalds. It is now a *free* implementation of UNIX available via ftp or on disk. I got *Linux Installation and Getting Started* after meeting Linus at the Boston USENIX. Matt Welsh, who is the coordinator of the Linux Documentation Project, has done a fine job. There are typos, but they aren't important. Like so many other user-projects, Linux is worth supporting; and this small book is worth reading.

More X than you wanted

When Xes on a piece of paper were kisses, I didn't mind getting them. I use the X Window System. I am currently running the Motif Window Manager. ORA's six-feet of manuals and Mansfield's *X Window System* (Addison-Wesley, 1991) are more than enough. A week ago, the mail brought me four new books on the topic from Prentice Hall. I skimmed the two by Young, which seem to be OK. Culwin's is actually interesting. Smith's is competent, respectable, and unnecessary. I don't know what gets into publishers. Given a topic, each of them has to have a book on it.

So far as I'm concerned, there's a limit to just how many books on a given topic the market can bear. X and its commercial descendants seem to have exhausted the number of books even fanatics will purchase. But then there aren't as many, yet, as there are with Internet in the title.

UNIX Commands

The UNIX and X Command Compendium contains over 2,000 entries covering "all commands, programs, and utilities" for X, OSF/Motif, OpenWindows, OpenLook, SunOS, Solaris, AIX, SCO UNIX, BSD, ``and others.'' It doesn't.

Southerton and Perkins have put together a useful compendium. But far from "all." As an exercise, I looked at the ORA *4.4 Programmer's Reference* volume. Missing from S&P are `accept(2)`, `access(2)`, `bind(2)`, `brk(2)`. I stopped at that point. I thought: maybe they mean all the user commands. So I got out

Books reviewed in this column:

Donnalyn Frey and Rick Adams, **!%@:: A Directory of Electronic Mail Addressing and Networks**. 4th Ed. O'Reilly & Associates, 1994. 640pp. ISBN 1-56592-046-5. $9.95.

CSRG, **4.4BSD documents set.** 5 vols. The USENIX Association and O'Reilly & Associates, 1994. ISBN 1-56592-082-1. $120.

CSRG, **BSD4.4-Lite CD-ROM Companion.** ISBN 1-56592-081-3. $36. [set plus disk: $150]

Matt Welsh, **Linux Installation and Getting Started.** SSC, 1994. 250pp. ISBN 0-916151-71-9. $12.95.

Douglas A. Young, **Motif Debugging and Performance Tuning**. Prentice Hall, 1994. 512pp. ISBN 0-13-147984-9. $36.

Douglas A. Young, **X Window System: Programming and Applications with Xt** 2nd Ed. Prentice Hall, 1994. 656pp. ISBN 0-13-123803-5. $36.

Fintan Culwin, **An X/Motif Programmer's Primer. 0** Prentice Hall, 1994. 344pp. ISBN 0-13-101841-8.

Jerry Smith, **X: A Guide for Users.** Prentice Hall, 1994. 339pp. ISBN 0-13-12379-5. $29.

Alan Southerton and Edwin C. Perkins, Jr., **The UNIX and X Command Compendium.** John Wiley & Sons, 1994. 640pp. ISBN 0-471-30982-6. $19.95.

Simson Garfinkel, Daniel Weise, and Steven Strassmann, **The UNIX-Haters Handbook.** IDG Books, 1994. 329pp. ISBN 1-56884-203-1. $16.95.

Daniel P. Petrozzo and John C. Stepper, **Successful Reengineering.** Van Nostrand Reinhold, 1994. 356pp. ISBN 0-442-01722-7. $24.95.

the brand-new *4.4 User's Reference*. Missing from S&P were `adb(1)`, `addftinfo(1)`, `afmtodit(1)`, and `ansitape(1)`, from among the a listings. The `-w` flag is missing from the material on `awk`. Complete it ain't.

Whining, not dining

While I'm being curmudgeonly (I'm taking lessons from Stan Kelly-Bootle), let me look at *UNIX-Haters*.

In 1981, Don Norman published "The Trouble with UNIX." In a previous incarnation, he did some of the best work I have ever read (e.g., *Memory and Attention*, 1969). A lot of what he wrote nearly 15 years ago made sense. But now that he's left UCSD for Apple, his plaintive introduction sounds strident. There are many things I am unhappy with where UNIX is concerned. There are even more things I hate about my Mac. And I'm not even going to tell IBM 360/370 stories. But I don't go around whining about the fact that I "really liked" my DECwriterII in 1975.

So it's unclear to me what the purpose of this book is. I know two of the authors personally. I admit to never having spoken to either about Operating Systems. But it is true that Michael Travers started a list called "unix-haters" in October 1987. So, in the tradition of books on the 100 worst books on (fill in) and *All I ever Needed to Know I learned from Star Trek*, here's what Garfinkel-Weise-Strassmann think is the "best of the unix-haters on-line mailing list." It is chock full of whines and silliness. I actually read most of it. You don't need to. Outside of the folks on the list, it's not clear to me who would.

The title page says, "Two of the most famous products of Berkeley are LSD and Unix. I don't think that is a coincidence." I presume the editors endorse this data-less remark of someone who had fried his brains. LSD was synthesized by a Swiss pharmacologist in the late 1930s. UNIX was born in New Jersey in 1969 and didn't get to Berkeley for nearly five years. This kind of accuracy is typical of most of the book. Of course, there were good things in TENEX and Multics. Some of them have been carried into more successful systems. Others haven't been. If you want 'em, port 'em yourself. The great thing about UNIX has always been that it let you do stuff.

Nostalgically, whining about the demise of your old beloved OS or radio or the loss of your favorite baseball glove or copy of *The Tower Treasure* just isn't worth my time. Oh, yeah. There are complaints about emacs and C++ here, too. So what. Use another editing environment. Use a different language. There's Eiffel. There's Smalltalk.

And I really miss my 1966 Mustang. It died in 1976 at over 200,000 km. At that time it was running on five and two half cylinders. One had been totally dead for about two years.

UNIX-Haters comes with a "barf bag." The book didn't fit into mine.

Business stuff

Well, what do Total Quality Management, Open Systems, Re-engineering, Client/Server, and POSIX have in common?

They're all buzzwords. Re-engineering recently succeeded "just-in-time" and "total quality." A management fad, *The Economist* has pointed out, like a love-affair, "goes through a fairly predictable cycle from infatuation to disillusionment."

When Petrozzo and Stepper arrived, I only knew that re-engineering had been conceived by Champy and Hammer in 1990. I read the book over a few days. It was, in fact, quite interesting reading. Then I was told that 85% of re-engineering projects fail. (That's an exaggeration.) The local business library came to my aid, and I looked at *State of Re-engineering Report 1994* from CSC Index, a leading re-engineering consultancy. At the same time that CSC admits that re-engineering is not a guarantee of corporate renewal, and plays down tremendously the facts of job loss, etc., it makes several admissions. To me, the most important one is that re-engineering is not enough on its own. It needs to be linked to strategy. Managers need to reflect.

I think folks should read Petrozzo and Stepper so that they'll understand what the corporate executives have bought. My favorite piece of advice is on p. 209, "Avoid reports more than a few pages long."

Shakespeare suggests that we "kill all the lawyers." *The Economist* suggests (July 2, 1994) that it's "clearly time to re-engineer the re-engineers." Reflect on that.

USENIX Symposium on Very High Level Languages (VHLL)

October 26 – 28, 1994
Santa Fe, New Mexico

The USENIX Symposium on Very High Level Languages will spotlight high level languages and their usefulness in leveraging specific problem areas. The Symposium will introduce participants to new concepts and approaches through original unpublished work. Programmers will learn about the relative strengths and weaknesses and extract the key concepts common to the languages presented.

Using very high level languages (VHLLs), programmers can assemble entire applications from large building blocks in just a small fraction of the time required if conventional programming strategies were used. Programmers take advantage of increasingly available hardware cycles, trading cheap machine time for costly programmer time. VHLLs offer one of the most promising approaches toward radically improving programmer productivity.

UNIX has long supported very high level languages – consider AWK and the various shells. Often programmers create new little languages whenever a problem appears of sufficient complexity to merit a higher level programming interface – consider sendmail.cf. In recent years many UNIX programmers have turned to VHLLs for rapid prototypes and complete applications. They take advantage of these languages' higher level of abstraction to complete projects more rapidly and easily than they could with lower level languages.

While VHLLs such as TCL, Perl, Icon, and REXX have gained widespread use and popularity, many others never see the public light. Some of these languages address a limited problem domain (such as graphics, text processing, or mathematical modeling) using powerful primitives created for that specific problem. Other VHLLs are more general-purpose, but still much higher level than most traditional compiled languages. Some are stand-alone languages, while others can be embedded in other programs. Many are interpreted, although some are compiled to native machine code; a few occupy a gap between both worlds.

Preliminary Technical Program

Wednesday, October 26

8:45-9:45
Dr. Jon Bentley, AT&T Bell Laboratories

Good languages get the job done; they are useful and clean, but they don't have fans. Great languages will inspire passionate users. This talk surveys some of the languages that I have loved, from AWK to Visual Basic. I will illustrate the languages with the kinds of programs I would like to see in documentation; tiny programs to display language features and small programs that solve substantial problems.

Jon L. Bentley is a Member of Technical Staff in the Computing Science Research Center at AT&T Bell Laboratories. His research interests include programming techniques and algorithm design. Dr. Bentley received a B.S. degree in Mathematical Sciences from Stanford University in 1974, and M.S. and Ph.D. degrees in Computer Science from the University of North Carolina in 1976. He is the author of three books: *Writing Efficient Programs*, *Programming Pearls*, and *More Programming Pearls*.

10:15-11:15 Language Overview: Perl
Larry Wall, NetLabs, Inc.

Originally perceived as a text-processing language for writing impenetrable one-liners, Perl has recently developed into a language that can be used in polite company. Larry Wall, the author of Perl, will talk about what happens when you try to combine all your favorite languages into one language. He'll present the original design rationale for Perl, and how "Perl philosophy" is evolving with the development of Perl version 5, and why you should care.

11:15-12:15 Language Overview: TCL – A Universal Scripting Language
Dr. John Ousterhout, Sun Microsystems, Inc.

In this talk I will give a brief overview of Tcl, a universal scripting language, and Tk, its companion GUI toolkit. Then I will discuss how the Tcl language evolved and the design issues behind it. Finally, I will critique the language and describe what I would do differently if I were to start again.

1:30-2:30 Language Overview: Python Programming Language
Guido van Rossum, CWI

Python is an interpreted, object-oriented language with a clear, intuitive syntax, powerful high-level data structures, and a flexible dynamic type system. It provides modules and classes which make the construction of large Python programs feasi-

ble. The talk will start with a quick introduction to Python, then discuss the rationale of its design, and round off with a look in the crystal ball.

2:30-3:30 Language Overview: REXX

Pamela J. Taylor, REXX Language Association, The Workstation Group

REXX is a versatile language used for applications that include "throw-away" procedures, "glue" programs, prototyping, systems administration, and mission-critical business applications. This presentation will discuss: The philosophy of the language and the history of its development; features that make it easy to learn, use and suited for a broad range of applications.

4:00-5:00 Language Overview: Icon Programming Languages

Dr. Clinton Jeffery, University of Texas – San Antonio

Icon is a general-purpose programming language derived from Snobol4. It's primary innovation is an expression evaluation model that integrates procedural programming with generators and backtracking. This is matched in utility by a large repertoire of built-in operations and heterogeneous structures. An optimizing compiler and portability to platforms ranging from supercomputers and IBM mainframes to many PC operating systems broaden Icon's appeal.

7:00-8:30pm Invited Talk: From Blazon to Postscript

Daniel V. Klein, LoneWolf Systems

8:30-10:00pm Invited Talk: Objecting to Objects

Stephen C. Johnson, Melismatic Software

Thursday, October 27

8:30-9:30 Language Overview: Standard ML

Andy Koenig, AT&T Bell Laboratories

Standard ML is a strongly typed general-purpose language with particularly good support for functional programming, data abstraction, and composition of modules. It feels a little like strongly typed Lisp with different syntax. A robust compiler that generates fast machine code is available free of charge.

10:00-11:30 SESSION 1

An Anecdote About ML Type Inference

Andy Koenig, AT&T Bell Laboratories

libscheme: Scheme as a C Library

Brent Benson Jr., Harris Computer Systems

A New Architecture for the Implementation of Scripting Languages

Adam Sah and Jon Blow, University of California, Berkeley

1:00-2:30 SESSION 2

Tcl/Tk for a Personal Digital Assistant

Karin Petersen, Xerox PARC

Using Tcl to Control a Computer-Participative Multimedia Programming Environment

Christopher Lindblad, Massachusetts Institute of Technology

TkPerl – A Port of the Tk Toolkit to Per15

Malcolm Beattie, Oxford University

3:00-4:30 SESSION 3

Rapid Programming with Graph Rewrite Rules

Andy Schuerr, Aachen University of Technology

End-User Systems, Reusability, and High-Level Design

John Snyder, Kiem-Phong Vo and Glenn Fowler, AT&T Bell Laboratories

Compiling Matlab

Stephen C. Johnson, Melismatic Software; Cleve Moler, The Mathworks, Inc.

4:30-5:30 Invited Talk: Lessons Learned from Postscript

Dick Dunn, QMS Inc.

6:00-8:00 USENIX Reception

Friday, October 28

8:30-10:00 SESSION 4

Ksh: An Extensible High-Level Language

David Korn, AT&T Bell Laboratories

Fornax: A General-Purpose Programming Language

J. Storrs Hall, Rutgers University

Graphics Programming in Icon Version 9

Clinton Jeffery, University of Texas – San Antonio; Ralph Griswold and Gregg Townsend, University of Arizona

10:30-11:30 SESSION 5

Application Languages in Software Production

David Ladd and Christopher Ramming, AT&T Bell Laboratories

Using a Very-High Level Language to Build Families of High Quality Reusable Components
Gary Pollice, CenterLine Software, Inc.

1:00-2:00 SESSION 6

Dixie: A Distributed Internet Execution Environment
R. Stockton Gaines, University of Southern California, Information Sciences Institute

Feature-Based Portability
Glenn Fowler, David Korn, John Snyder and Kiem-Phong Vo, AT&T Bell Laboratories

2:00-3:00 Footnote: High-Level Languages, Little Languages, and Software Productivity
Stephen C. Johnson, Melismatic Software

Traditional methods of writing software are pricing themselves out of the market. For a while, traditional software methods will survive through offshore manufacturing, but the future belongs to high level languages both general purpose and special purpose ("little languages") – that can exploit cheap machine cycles to replace expensive programmers, shorten design cycles, and even lead to "user configurable" software.

3:00 Closing Remarks
Tom Christiansen, Program Chair

Program Committee:

Program Chair: Tom Christiansen, *Consultant*
Jon Bentley, *AT&T Bell Laboratories*
Stephen C. Johnson, *Melismatic Software*
Brian Kernighan, *AT&T Bell Laboratories*
John Ousterhout, *University of California, Berkeley*
Henry Spencer, *University of Toronto*
Larry Wall, *NetLabs, Inc.*

Registration Information

For complete symposium information, you can access the USENIX Resource Center on the World Wide Web: *http://usenix.org* or contact the USENIX conference office at 1 714 588 8649 or via email at <*conference@usenix.org*>.

First Symposium on Operating Systems Design and Implementation (OSDI)

November 14-17, 1994
Monterey, California

Sponsored by the USENIX Association, and Co-sponsored by ACM SIGOPS and IEEE TCOS

The OSDI symposium presents some of the best new research in operating and distributed systems: out of 178 submitted papers, the authors of the top 21 will present their work.

Six tutorials offer more reflective and in-depth analysis by experts on current systems and issues. Their topics include three microkernel-based and object-oriented systems, distributed and fault tolerant communication, both message and memory based, and structuring network code to attain very high speeds.

How can an operating system adapt to the widely varying needs of different applications, domains, and environments? During OSDI a panel of prominent researchers will discuss their current work in creating radically new OS architectures that address this problem, and provide a perspective on the field. Ample time will be provided for audience participation.

Authors of important new work in the Mach and CHORUS operating systems will present their results in an afternoon workshop following the last regular OSDI session. Most will have technical reports available for distribution to attendees.

Other attractions during OSDI are an evening panel on some controversial issue; Birds-of-a-Feather sessions on Mach, CHORUS, Spring and whatever other topics attendees wish; fifteen selected works-in-progress; a well-known keynote speaker to be announced; and finally, the lovely Monterey Bay location.

Important Dates

Hotel Reservation Deadline: October 21, 1994
Pre-Registration Discount Deadline: October 31, 1994

Program Committee

Jay Lepreau, Program Chair, *University of Utah*
Brian Bershad, *University of Washington*
David Black, *OSF Research Institute*
Paul Leach, *Microsoft Corporation*
Jim Lipkis, *Chorus Systèmes*
Karin Petersen, *Xerox PARC*
Larry Peterson, *University of Arizona*

Karsten Schwan, *Georgia Institute of Technology*
Michael Scott, *University of Rochester*
Willy Zwaenepoel, *Rice University*

Tutorials

Monday, November 14

The Spring Operating System: Internals Overview, *Thomas W. Doeppner, Brown University*

Reliable Distributed Computing Using the Isis and Horus Systems, *Ken Birman, Cornell University*

The Architecture of the GNU Hurd, *Michael Bushnell, Free Software Foundation*

The Architecture of CHORUS, *Jim Lipkis, Chorus Systèmes*

Distributed Shared Memory: Principles, Practices, and Packages, *John Carter, University of Utah*

The x-kernel: OS Support for High-Speed Networking *Larry Peterson, University of Arizona*

Advance Technical Program

Tuesday, November 15

Opening Remarks and Keynote 9:00 - 10:30
Opening Remarks, *Jay Lepreau, University of Utah*
Keynote address – TBA

Scheduling and Mobility 11:00 - 12:30

Lottery Scheduling: Flexible Proportional-Share Resource Management, *Carl A. Waldspurger, William E. Weihl, MIT*

Scheduling for Reduced CPU Energy, *Mark Weiser, Al Demers, Brent Welch, Scott Shenker, Xerox PARC*

Storage Alternatives for Mobile Computers, *Fred Douglis, Ramon Caceres, Frans Kaashoek, Kai Li, Brian Marsh, Joshua Tauber, MITL*

File Systems 2:00 - 3:30

Opportunistic Log: Efficient Installation Reads in a Reliable Storage Server, *James O'Toole, Liuba Shrira, MIT*

Metadata Update Performance in File Systems, *Gregory R. Ganger, Yale N. Patt, University of Michigan*

Disk-directed I/O for MIMD Multiprocessors, *David Kotz, Dartmouth College*

Distributed Shared Memory I 4:00 - 5:30

Integrating Message-Passing with Lazy Release Consistent Distri uted Shared Memory, *Povl T. Koch, Robert J. Fowler, University Copenhagen*

Software Write Detection for Distributed Shared Memory, *Matthe J. Zekauskas, Wayne A. Sawdon, Brian N. Bershad, Carnegie Me lon University*

The Design and Evaluation of a Shared Object System for Distril uted Memory Machines, *Daniel J. Scales, Monica S. Lam, Stanfo University*

Birds-of-a-Feather Sessions 9:00 - 11:00

Wednesday, November 16

Networking and Multiprocessing 9:00 - 10:30

PATHFINDER: A Pattern-Based Packet Classifier, *Mary L. Baile Burra Gopal, Michael A. Pagels, Larry L. Peterson, Prasenjit Sarkar, University of Arizona*

Performance Issues in Parallelized Network Protocols, *Erich M. Nahum, David J. Yates, James F. Kurose, Don Towsley, Universi of Massachusetts*

Experiences with Locking in a NUMA Multiprocessor Operating System Kernel, *Ronald C. Unrau, Orran Krieger, Benjamin Gamsa, Michael Stumm, University of Toronto*

Works-in-Progress 11:00 - 12:30
Fifteen 5-minute presentations. Submit your abstract to <*osdi-wip@cs.utah.edu*> by Wednesday, November 9, 5pm MST

Steps to Extensibility 2:00 - 3:30

HiPEC: High Performance External Virtual Memory Caching *Chao Hsien Lee, Meng Chang Chen, Ruei Chuan Chang, Natior Chiao Tung University*

Implementation and Performance of Application-Controlled File Caching, *Pei Cao, Edward W. Felten, Kai Li, Princeton Univers*

A Caching Model of Operating System Kernel Functionality *David R. Cheriton, Kenneth J. Duda, Stanford University*

Panel: Radical OS Structures for Extensibility 4:00 - 5:30 Moderator: Paul Leach

Invited panelists will present their architectures and provide per spective on the issues. Attendees with designs for their own exte sible OSs are invited to bring technical reports for distribution, a along with all attendees, to speak at the floor microphone.

Panel: "Controversial Topic" 8:00

Thursday, November 17

Distributed Shared Memory II 9:00 - 10:30

Distributed Filaments: Efficient Fine-Grain Parallelism on a Cluster of Workstations, *Vincent W. Freeh, David K. Lowenthal, Gregory R. Andrews, University of Arizona*

Integrating Coherency and Recovery in Distributed Systems *Michael J. Feeley, Jeffrey S. Chase, Vivek R. Narasayya, Henry M. Levy, University of Washington*

Garbage Collection and DSM Consistency, *Paulo Ferreira, Marc Shapiro, INRIA and Universite Pierre et Marie Curie*

Memory Management 11:00 - 12:30

Software Prefetching and Caching for Translation Lookaside Buffers, *Kavita Bala, M. Frans Kaashoek, William E. Weihl, MIT*

Dynamic Page Mapping Policies for Cache Conflict Resolution on Standard Hardware, *Theodore H. Romer, Dennis Lee, Brian N. Bershad, University of Washington*

Cooperative Caching: Using Remote Client Memory to Improve File System Performance, *Michael D. Dahlin, Thomas E. Anderson, David A. Patterson, Randolph Y. Wang, University of CA, Berkeley*

Mach/CHORUS Workshop: Tracing and Performance 2:00 - 3:30

Micro-Kernel Support for Trace-Replay, *Frederic Ruget, Chorus Systèmes and Unite mixte Bull-IMAG, Universite Joseph Fourier*

Concurrent Remote Task Creation, *Dejan Milojicic, David Black and Steve Sears - OSF Research Institute*

Microkernel Modularity with Integrated Kernel Performance, *Michael Condict, Don Bolinger, Dave Mitchell, and Eamonn McManus, OSF Research Institute*

Mach/CHORUS Workshop: (Continued) 4:00 - 5:30

TBA: Three Mach/CHORUS papers will be presented

For complete conference information you can access the USENIX Resource center on the World Wide Web: *http://usenix.org* or contact the USENIX Conference Office at 1 714 588 8649 or via email at <*conference@usenix.org*>.

SANS-IV: Tools and Techniques You Can Use Immediately

April 24-29, 1995
Washington, DC

Call for Abstracts

Dates For Refereed Papers Submission:

- Extended abstracts due: November 1, 1994
- Notification to authors: December 1, 1994
- Camera-ready final papers due: February 1, 1995

Program Committee:

Program Chair: Rob Kolstad, *Berkeley Software Design*
Tom Barrett, *Pacific Bell*
Matt Bishop, *UC Davis*
Tom Christiansen, *Perl Wizard*
Michele Crabb, *NASA Ames*
William Howell, *Glaxo Pharmaceuticals*
E. Scott Menter, *Enterprise Systems Management Corp.*
Alan Paller, *Computer Associates*
Dale Pfaff, *US Naval Research Laboratory*
Marcus Ranum, *Trusted Information Systems*
Bjorn Satdeva, *sys/admin, inc.*
Peg Schafer, *BBN*
Ace Stewart, *NASA Ames*
Dave Taber, *SunSoft*
Pat Wilson, *Dartmouth College*
Elizabeth Zwicky, *SRI International*

Sponsors: The Open Systems Conference Board in cooperation with the USENIX Association's Special Technical Group SAGE (The System Administrators Guild), and the Washington Area UNIX Users Group.

SANS is the annual spring conference combining system administration, network management, and security. It offers three days of in-depth, authoritative courses and two days of multiple tracks of invited and peer-reviewed technical papers. It provides a forum in which system administrators, network managers, and security experts can exchange practical information, share new ideas, and evaluate new tools.

SANS-IV continues the tradition of focusing exclusively on practical solutions to today's problems – providing techniques you can put to work immediately. It also provides a unique opportunity to review the most popular commercial software tools and focuses on how those tools can lower the costs of managing UNIX and client/server computing, and includes topical Birds Of A Feather sessions and special events. Formally reviewed papers, presented at the conference, will be published in the conference proceedings.

Conference Topics

The overall focus of **SANS IV** is finding practical solutions to common problems in systems, security, and network administration. Feel free to offer abstracts on any of the following topics:

General Management Topics:

User Education Techniques
Techniques For Dealing With Problem Users
Managing Your Boss
Politics Of Systems Administration
Helping Mainframe Operations Staff Transition To UNIX
Special Challenges Of Administration at Military Sites

System Administration Topics:

Making Backup Less Painful
Mail Handling and Automation
Managing Heterogeneous Systems
Lights-Out Operations
System Scheduling and Monitoring
Better Storage Solutions
Accounting and Chargeback
Off-The-Shelf Tools(any experiences with them)
Tools You Don't Like and Why
Tools You DO Like and Why
Tools You Created
Distributed Administration, Including OSF's DME
Planning for Manageable Systems
Managing UNIX along with Windows PCs or Macs
Managing UNIX along with MVS, VMS, or OS/400

Security Topics:

Firewalls
Security and The Network
Security for Internet
Kerberos and other Authentication Techniques
Badge Readers, Finger Print Readers and Other Physical Security Devices
Security Incidents
Setting Up and Working With Emergency Response Teams
Security Tools From The Net: How To Find Them and How To Use Them
Commercial Security Tools
Can and Should the Superuser Be Controlled?
Politics of Security; Gaining Top Management Support
Auditing and Monitoring; Integration With Network Monitors

Network Management Topics:

Managing Heterogeneous Networks
Using SNMP
Network Security Monitoring
Network Monitoring And Performance Testing
Training And Education
Techniques For Dealing With Remote Users
Networked Backup Schemes
Distributed Mail Systems
Domain Name Service Configuration
Centralizing Message Monitoring For Heterogeneous Systems
OSF's DCE
Off-the-Shelf Tools (any experiences with them)
Tools You Do or Don't Like And Why (SunNetManager, OpenView, NetView/6000 or others)

You don't have to have made a major breakthrough to have your paper accepted. The delegates just want practical solutions for real problems.

Abstract Submission

Extended abstracts must be 2 to 5 pages long and be received by November 1, 1994. The object of an extended abstract is to convince the reviewers that a good paper and presentation will result. Your abstract should include:

- A cover page including the title of the paper, the principal author's name, address, email, telephone, and FAX numbers, and the names of the other authors.
- A description of the problem and its importance.
- Your solution, including details of how it worked. If this is work on emerging technology, show what the expected impact will be. If your solution is based on commercial hardware or software tools, name them. (Abstracts from software vendors are also welcome, and will be considered as part of the tools track or the regular paper sessions, depending on their focus.)
- Data on how well the solution works: before/after comparisons, direct savings, trade-offs, etc.
- Lessons learned.

Where to submit:

Please send one copy of your extended abstract to the program committee using one of the following methods. All submissions will be acknowledged.

Preferred method: email (plain text) to <*sans@fedunix.org*>
Alternative method: postal delivery (on paper) to SANS-IV Abstracts, 4610 Tournay Road Bethesda, MD 20816
Questions: email <*sans@fedunix.org*> or telephone 719-599-4303

ANNOUNCEMENT & PRELIMINARY CALL FOR PAPERS

2nd USENIX Symposium on Mobile and Location-Independent Computing

APRIL 10–11, 1995
ANN ARBOR, MICHIGAN

The Second USENIX Symposium on Mobile and Location-Independent Computing will provide a major opportunity for researchers and practitioners in this rapidly growing field to exchange ideas and present results of their work.

The First Mobile Computing Symposium, held in Boston in August 1993, generated a great deal of interest from the UNIX and mobile computing communities. Since that time, mobile computing has become an even hotter topic, with the size, cost, and power requirements of the equipment going down. The FCC has announced a plan to auction radio spectrum for use of mobile devices, and the Internet Engineering Task Force (IETF) is in the process of standardizing protocols for mobile TCP/IP, including roaming capabilities. Mobile computers are the fastest growing segment of the PC market, airlines are scrambling to provide network connectivity on board, and terminal rooms at computer conferences routinely provide network taps for users who bring their own computers.

The 1995 symposium is a single-track symposium offering two days of refereed paper presentations. The symposium will also include panels, Work-in-Progress reports, Birds-of-a-Feather sessions, and a Keynote speaker. Formally reviewed papers, presented during the symposium, will be published in the symposium proceedings. Proceedings will be distributed free to attendees and later will be available for purchase from the USENIX Association.

Dates For Refereed Paper Submissions

- Extended abstracts due: January 2, 1995
- Notification to authors: January 23, 1995
- Camera-ready final papers due: March 6, 1995

Program Committee

- Program Chair: Jim Rees, University of Michigan
  Jim.Rees@umich.edu

Dan Duchamp, Columbia University
djd@cs.columbia.edu
Dan Geer, OpenVision Technologies
geer@cam.ov.com
Phil Karn, Qualcomm
karn@qualcomm.com
Jim Kempf, Sun Microsystems
james.kempf@eng.sun.com
Jay Kistler, Digital Equipment Corporation
jjk@src.dec.com

Symposium Topics

We seek original and innovative papers about current developments in mobile and location-independent computing. We are especially interested in reports on practical experiences with mobile systems. The Mobile Computing Symposium will address a wide range of issues and ongoing developments, including, but not limited to:

- Applications for the mobile user
- Navigation and positioning (GPS, etc.)

- Security, especially in wireless environments or when away from home
- Caching and disconnected operation of applications and file systems
- Communications Protocols, including mobile TCP/IP
- Wireless communications (CDPD, CDMA, GSM, Ardis/RAM, cellular modem, etc.), and how they relate to and interact with operating systems and applications
- Portable and mobile computing equipment

Refereed Paper Submissions

Submission of an extended abstract of 1500–2500 words (9000–15000 bytes or 3–5 pages) is recommended. Shorter abstracts run a significant risk of rejection as there will be little on which the program committee can base an opinion. Extended abstracts should be sent to Jim Rees at the address below.

If you would like to receive detailed guidelines for submission and examples of extended abstracts, you may telephone the USENIX Association office at +1-510-528-8649 or email to *mobile2authors@usenix.org*.

For administrative reasons (not blind reviewing), each submission should include a separate page or e-mail message giving the title of the paper, the names and affiliations of the authors, and the name of the author who will act as the contact person for the program committee. For the contact person, also include a daytime telephone number, postal address, e-mail address and FAX number if possible.

USENIX symposia, like most symposia and journals, require that papers not be submitted simultaneously to more than one conference or publication and that submitted papers not be previously or subsequently published elsewhere. Papers accompanied by "non- disclosure agreement" forms are not acceptable and will be returned to the author(s) unread. All submissions are held in the highest confidentiality prior to publication in the Proceedings, both as a matter of policy and in accord with the U.S. Copyright Act of 1976.

For More Program Information

For questions about refereed paper submissions and other program concerns, contact the Program Chair:

- Jim Rees
  CITI
  University of Michigan
  519 West William
  Ann Arbor, Michigan 48103 USA
- Internet: *Jim.Rees@umich.edu*
- Telephone: +1-313-763-4174
- Fax: +1-313-763-4434

Dates For Refereed Paper Submissions

- Extended abstracts due: January 2, 1995
- Notification to authors: January 23, 1995
- Camera-ready final papers due: March 6, 1995

For Registration Information

Materials containing all details of the technical and tutorial programs, registration fees and forms, and hotel information will be available beginning in February 1995. If you wish to receive the registration materials, please contact USENIX at:

- USENIX Conference Office
  22672 Lambert Street
  Suite 613
  Lake Forest, CA 92630
  USA
- Telephone: +1-714-588-8649
- Fax: +1-714-588-9706
- Internet: *conference@usenix.org*

5th USENIX UNIX Security Symposium

JUNE 5–7, 1995
MARRIOTT HOTEL
SALT LAKE CITY, UTAH

ANNOUNCEMENT AND PRELIMINARY CALL FOR PAPERS

SPONSORED BY THE USENIX ASSOCIATION, IN COOPERATION WITH: THE COMPUTER EMERGENCY RESPONSE TEAM (CERT), IFIP WG 11.4, AND UNIFORUM

OVERVIEW

The goal of this symposium is to bring together security practitioners, researchers, system administrators, systems programmers, and others with an interest in computer security as it relates to networks and the UNIX operating system.

This will be a three day, single-track symposium consisting of tutorials, refereed and invited technical presentations, and panel sessions. The first day will be devoted to tutorial presentations. Two days of technical sessions will follow the tutorials.

TUTORIALS

♦ **June 5**

This one-day tutorial program is designed to address the needs of both technical and management attendees. The tutorials will supply overviews of various security mechanisms and policies. Each will provide specifics to the system and site administrator for implementing numerous local and network security precautions, firewalls, and monitoring systems.

KEYNOTE AND TECHNICAL SESSIONS

♦ **June 6–7**

The keynote address by Stephen T. Walker, founder and president of Trusted Information Systems, will begin the technical sessions program. Mr. Walker will speak on information security and privacy in computing. Mr. Walker is an electronics engineer and computer systems analyst with over 25 years of experience in system design and program management; particularly extensive is his experience with the design and implementation of large scale computer networks and information systems. He is nationally recognized for his pioneering work on the DoD Computer Security Initiative, the establishment of the National Computer Security Center, and the formation of the Defense Data Network. He is a member of the Computer System Security and Privacy Advisory Board, established by the Computer Security Act of 1987.

The technical sessions program, in addition to presentations of refereed papers, will include invited talks, and possibly panel sessions. There will also be two evenings available for Birds-of-a-Feather sessions (BoFs) and Works-in-Progress Reports (WiPs). The program committee invites you to submit proposals, ideas, or suggestions for these presentations; your suggestions may be submitted to the program chair via email to *security@usenix.org* or by post to the address given on the following page.

The program committee will formally review and accept papers for presentation at the symposium and publish them in the symposium proceedings. USENIX will provide provide the proceedings free to technical session attendees; additional copies will be available for purchase from USENIX.

IMPORTANT DATES

DATES FOR REFEREED PAPER SUBMISSIONS:

- Extended abstracts due: February 13, 1995
- Program Committee decisions made: March 8, 1995
- Camera-ready final papers due: May 1, 1995
- Registration materials available: March 1995

USENIX®

The UNIX® and Advanced Computing Systems Professional and Technical Association.

Symposium Topics

Presentations are being solicited in areas including but not limited to:

- User/system authentication
- File system security
- Network security
- Security and system management
- Security-enhanced versions of the UNIX operating system
- Security tools
- Security incident investigation and response
- Computer misuse and anomaly detection

Refereed Paper Submissions

Submissions must be received by February 13, 1995. Full papers should be 10–15 pages. Instead of a full paper, authors may submit an extended abstract which discusses key ideas. Extended abstracts should be 5–7 pages long (about 2500–3500 words), not counting references and figures. The body of the extended abstract should be in complete paragraphs. The object of an extended abstract is to convince the reviewers that a good paper and presentation will result. All submissions will be judged on originality, relevance, and correctness.

An individual program committee member will be assigned to shepherd each accepted submission through preparation of the final paper. The assigned member will act as a conduit for feedback from the committee to the authors. Camera-ready final papers are due May 1, 1995.

Please accompany each submission by a cover letter stating the paper title and authors along with the name of the person who will act as the contact to the program committee. Please include a surface mail address, daytime and evening phone number, and, if available, an email address and fax number for the contact person.

If you would like to receive detailed guidelines for submission and examples of extended abstracts, you may telephone the USENIX Association office at +1 510 528 8649, or email to *securityauthors@usenix.org* or to the program chair.

The USENIX UNIX Security conference, like most conferences and journals, requires that papers not be submitted simultaneously to another conference or publication and that submitted papers not be previously or subsequently published elsewhere. Papers accompanied by "non-disclosure agreement" forms are not acceptable and will be returned to the author(s) unread. All submissions are held in the highest confidentiality prior to publication in the Proceedings, both as a matter of policy and in accord with the U.S. Copyright Act of 1976.

Where To Submit

Please send one copy of a full paper or an extended abstract to the program committee via one of the following methods. All submissions will be acknowledged.

- Preferred method: email (PostScript or ASCII) to *securitypapers@usenix.org*
- Alternate method: postal delivery to Fred Avolio, Trusted Information Systems, 3060 Washington Road, Glenwood, MD 21738
- Phone: +1 301 854 6889
- Fax: +1 301 854 5363

Program Committee

- **Program Chair:** Fred Avolio, *Trusted Information Systems, Inc.*

Steve Bellovin, *AT&T Bell Laboratories*

Bill Cheswick, *AT&T Bell Laboratories*

Ed DeHart, *CERT*

Ed Gould, *Digital Equipment Corporation*

Marcus Ranum, *Trusted Information Systems, Inc.*

Jeff Schiller, *MIT*

Gene Spafford, *COAST Laboratory, Purdue University*

For Registration Information

Materials containing all details of the technical and tutorial programs, registration fees and forms, and hotel information will be available beginning in April 1995. If you wish to receive the registration materials, please contact USENIX at:

- USENIX Conference Office
  22672 Lambert Street
  Suite 613
  Lake Forest, CA 92630
  USA

Phone: +1 714 588 8649
Fax: +1 714 588 9706
Email: *conference@usenix.org*

For more information about USENIX and its events, access the USENIX Resource Center on the World Wide Web. The URL is *http:// www.usenix.org*. Or send email to our mailserver at: *info@usenix.org*. Your message should contain the line: *send catalog*. A catalog will be returned to you.

USENIX Conference on Object-Oriented Technologies (COOTS)

JUNE 26–29, 1995
MONTEREY, CALIFORNIA

ANNOUNCEMENT & PRELIMINARY CALL FOR PAPERS

The COOTS conference is designed to be a showplace for advanced development work in object-oriented technologies. The conference will emphasize research and experience derived from efforts to use object-oriented techniques to builds complex systems that meet real world needs.

The COOTS conference will begin with two days of tutorials. The tutorial program will offer a selection of tutorials from among several tracks. We expect tutorial topics to include:

- distributed object systems (CORBA, etc.)
- object-oriented network programming
- alternative object-oriented languages
- advanced techniques in memory management
- efficient and effective class design

Two days of technical sessions will follow the tutorials. Proceedings of the conference will be published by USENIX and will be provided free to technical session attendees; additional copies will be available for purchase from USENIX.

Like the USENIX C++ Conferences and Advanced Topics Workshops from which it is derived, COOTS will emphasize the advanced engineering aspects of object technology. While papers covering work in C++ are encouraged, the conference is broader in scope than its ancestors and invites submissions describing results and work in other object-oriented or object-based languages.

Dates For Refereed Paper Submissions

- Submissions due: March 6, 1995
- Notification to authors: April 3, 1995
- Camera-ready final papers due: May 15, 1995

Preliminary Program Committee

- Program Chair: Vincent F. Russo, *Purdue University*
- Tutorial Program Chair: Doug Lea, *SUNY Oswego*

Mark Linton, *Silicon Graphics*
Chris Pettus, *Taligent*
Jim Waldo, *SUN Microsystems*
Murthy Devarokonda, *IBM Watson Research Labs*
Addtional members to be announced.

Conference Topics

We seek papers describing original work concerning the design, implementation, and use of object-oriented technologies. Questions regarding a topic's relevance may be addressed to the program chair via electronic mail to *russo@cs.purdue.edu*.

Potential topics include:

- work on object-oriented programming languages (C++, Modula-3, Eiffel, etc.)
- implementations of commercial object infrastructures (CORBA, NextStep, OLE-II, SOM/DSOM, etc.)
- interface description languages
- distributed object systems
- unique applications of and experiences with object-oriented technologies

Refereed Paper Submissions

Submissions must be received by March 6, 1995. Full papers should be 10 to 15 pages. Instead of a full paper, authors may submit an extended abstract which discusses key ideas. Extended abstracts should be 5–7 pages long (about 2500–3500 words), not

counting references and figures. The body of the extended abstract should be complete paragraphs. The object of an extended abstract is to convince the reviewers that a good paper and presentation will result. While, by acceptance of extended abstracts, we intend to stimulate industrial participation, submission of extended abstracts by academics is in no way discouraged.

All submissions will be judged on originality, relevance, and correctness. Each accepted submission will be assigned a member of the program committee to act as its shepherd through the preparation of the final paper. The assigned member will act as a conduit for feedback from the committee to the authors. Camera-ready final papers are due May 15, 1995.

Please accompany each submission with a cover letter stating the paper title and authors along with the name of the person who will act as the contact to the program committee. Please include a surface mail address, daytime and evening phone number, and, if available, an email address and fax number for the contact person.

If you would like to receive detailed guidelines for submission and examples of extended abstracts, you may telephone the USENIX Association office at +1-510-528-8649, or email to *cootsauthors@usenix.org* or to the program committee chair.

The COOTS conference, like most conferences and journals, requires that papers not be submitted simultaneously to another conference or publication and that submitted papers not be previously or subsequently published elsewhere. Papers accompanied by "non-disclosure agreement" forms are not acceptable and will be returned to the author(s) unread. All submissions are held in the highest confidentiality prior to publication in the Proceedings, both as a matter of policy and in accord with the U.S. Copyright Act of 1976.

Where To Submit

Please send one copy of a full paper or an extended abstract to the program committee via one of the following methods. All submissions will be acknowledged.

- Preferred Method: email (Postscript or ASCII) to *cootspapers@usenix.org*
- Alternate Method: postal delivery to
  USENIX COOTS Conference
  c/o Dr. Vincent F. Russo
  Department of Computer Sciences
  Purdue University
  West Lafayette, IN 47907 USA
  Telephone: +1-317-494-6008

Dates For Refereed Paper Submissions

- Submissions due: March 6, 1995
- Notification to authors: April 3, 1995
- Camera-ready final papers due: May 15, 1995

For Registration Information

Materials containing all details of the technical and tutorial programs, registration fees and forms, and hotel information will be available beginning in April 1995. If you wish to receive the registration materials, please contact USENIX at:

- USENIX Conference Office
  22672 Lambert Street
  Suite 613
  Lake Forest, CA 92630
  USA
- Telephone: +1-714-588-8649
- Fax: +1-714-588-9706
- Internet: *conference@usenix.org*

9th USENIX Systems Administration Conference (LISA IX)

SEPTEMBER 18–22, 1995
MARRIOTT HOTEL
MONTEREY, CALIFORNIA

Important Dates

REFEREED PAPER SUBMISSIONS:
- Extended abstracts due: May 1, 1995
- Notification to authors: June 5, 1995
- Final papers due: August 1, 1995

REGISTRATION MATERIALS AVAILABLE:
- July, 1995

Co-sponsored by

USENIX®
The UNIX® and Advanced Computing Systems Professional and Technical Association

AND

SAGE
THE SYSTEM ADMINISTRATORS GUILD

The USENIX Systems Administration (LISA) Conference is widely recognized as the leading technical conference for system administrators. Historically, LISA stood for "Large Installation Systems Administration," back in the days when having a large installation meant having over 100 users, over 100 systems, or over one gigabyte of disk storage. Today, the scope of the LISA conference includes topics of interest to system administrators from sites of all sizes and kinds. What the conference attendees have in common is an interest in solving problems that cannot be dealt with simply by scaling up well-understood solutions appropriate to a single machine or a small number of workstations on a LAN.

The theme for this year's conference is "New Challenges," which includes such emerging issues as integration of non-UNIX and proprietary systems and networking technologies, distributed information services, network voice and video teleconferencing, and managing very complex networks. We are particularly interested in technical papers that reflect hands-on experience, describe fully implemented and freely distributable solutions, and advance the state of the art of system administration as an engineering discipline.

Tutorial Program

Monday and Tuesday, September 18–19, 1995

The two-day tutorial program offers up to five tracks of full and half-day tutorials. Tutorials offer expert instruction in areas of interest to system administrators of all levels, from novice through senior. Topics are expected to include networking, advanced system administration tools, Solaris and BSD administration, Perl programming, firewalls, NIS, DNS, Sendmail, and more.

To provide the best possible tutorial offerings, USENIX continually solicits proposals for new tutorials. If you are interested in presenting a tutorial at this or other USENIX conferences, please contact the tutorial coordinator, Daniel V. Klein:
- Phone: +1 412 421 0285; FAX: +1 412 421 2332; E-mail: *dvk@usenix.org*

Technical Sessions

Wednesday through Friday, September 20–22, 1995

The three days of technical sessions consist of two parallel tracks. The first track is dedicated to presentations of refereed technical papers. The second track will accommodate invited talks, panels and Works-in-Progress (WIP) sessions.

Conference Topics

Papers addressing the following topics are particularly timely; papers addressing other technical areas of general interest are equally welcome.
- Dealing with differences in UNIX implementations – migration and interoperability among BSD, SVR4, OSF and others
- Integration of UNIX-based with non-UNIX-based and proprietary systems and networking technologies (Mac, NT and DOS PCs)
- Application of emerging technologies (Mbone, Mosaic) to system administration
- Administration and security of distributed information services (WAIS, gopher, WWW) and network voice and video teleconferencing (Mbone)
- Experience supporting mobile and location-independent computing
- Experience with large (1000+ machine) networks, especially networks of SVR4-based systems
- Real-world experience with implementations of proposed system administration standards
- Unusual applications of commercial system administration software packages
- Application of operational planning techniques to system administration including measurements and metrics, continuous process improvement, automation, and increasing productivity
- File migration, archival storage & backup systems in extremely large environments
- Innovative tools and techniques that have worked for you
- Managing high-demand and high-availability environments
- Migrating to new hardware and software technologies

- Administration of remote sites that have no technical experts
- Supporting MIS organizations on UNIX
- Real-world experiences with emerging procedural/ethical issues – e.g., developing site policies, tracking abusers, and implementing solutions to security problems
- Networking non-traditional sites (libraries, museums, K–12)

Refereed Paper Submissions

An extended abstract is required for the paper selection process. Full papers are not acceptable at this stage; if you send a full paper, you must also include an extended abstract. "Extended" means 2–5 pages.

Include references to establish that you are familiar with related work, and, where possible, provide detailed performance data to establish that you have a working implementation or measurement tool.

Submissions will be judged on the quality of the written submission, and whether or not the work advances the state of the art of system administration. For more detailed author instructions and a sample extended abstract, send e-mail to: *lisa9authors@usenix.org* or call the USENIX office at +1 510 528 8649.

Note that USENIX, like most conferences and journals, requires that papers not be submitted simultaneously to more than one conference or publication and that submitted papers not be previously or subsequently published elsewhere. Papers accompanied by "non-disclosure agreement" forms are not acceptable and will be returned unread. All submissions are held in the highest confidence prior to publication in the conference proceedings, both as a matter of policy and as protected by the U.S. Copyright Act of 1976.

Authors of an accepted paper must provide a final paper for publication in the conference proceedings. At least one author of each accepted paper presents the paper at the conference. Final papers are limited to 20 pages, including diagrams, figures and appendixes, and must be in troff, ASCII, or LaTeX format. We will supply you with instructions. Papers should include a brief description of the site, where appropriate.

Conference proceedings, containing all refereed papers and materials from the invited talks, will be distributed to attendees and will also be available from USENIX following the conference.

Where to Send Submissions

Please submit extended abstracts for the refereed paper track by two of the following methods. All submissions will be acknowledged.

- E-mail to: *lisa9papers@usenix.org*; FAX to: +1 510 548 5738; Mail to: LISA 9 Conference, USENIX Association, 2560 Ninth Street, Suite 215, Berkeley CA USA 94710

To discuss potential submissions, and for inquiries regarding the content of the conference program, contact the program co-chairs at *lisa9chair@usenix.org* or at:

- Tina M. Darmohray, Lawrence Livermore National Laboratory, PO Box 808 L-510, Livermore CA USA 94550. +1 510 423 5999; FAX: +1 510 422 7869; E-mail: *tmd@usenix.org*
- Paul Evans, Synopsys, Inc., 700 East Middlefield Road, Mountain View CA USA 94043. +1 415 694 1855; FAX: +1 415 965 8637; E-mail: *ple@usenix.org*

Invited Talks Track

If you have a topic of general interest to system administrators, but that is not suited for a traditional technical paper submission, please submit a proposal for a second track presentation to the invited talk (IT) coordinators:

- Laura de Leon, Hewlett-Packard. +1 415 857 5605; FAX: +1 415 857 5686; E-mail: *deleon@hpl.hp.com*
- Peg Schafer, BBN. +1 617 873 2626; FAX: +1 617 873 4265; E-mail: *peg@bbn.com*

Program Committee

- Program Co-chair: Tina Darmohray, *Lawrence Livermore National Laboratory*
- Program Co-chair: Paul Evans, *Synopsys, Inc.*

Paul Anderson, *University of Edinburgh*
Kim Carney, *Massachusetts Institute of Technology*
Rob Kolstad, *Berkeley Software Design, Inc.*
Bryan McDonald, *SRI International*
Marcus Ranum, *Trusted Information Systems, Inc.*
John Schimmel, *Silicon Graphics, Inc.*

For Registration Information

All details of the technical and tutorial programs, registration fees and forms, and hotel information will be available in July, 1995. If you wish to receive the registration materials, please contact USENIX:

- USENIX Conference Office
  22672 Lambert Street, Suite 613
  Lake Forest, CA 92630 USA

Phone: +1 714 588 8649
Fax: +1 714 588 9706
E-mail: *conference@usenix.org*

For more information about USENIX and its events, access the USENIX Resource Center on the World Wide Web. The URL is *http://www.usenix.org*. Or send e-mail to our mailserver at: *info@usenix.org*. Your message should contain the line: *send catalog*. A catalog will be returned to you.

Vendor Display

Wed. & Thurs., Sept. 20–21, 1995

Well-informed vendor representatives will demonstrate products and services at the informal display. If your company would like to participate, please contact:

- Zanna Knight, +1 510 528 8649
  FAX: +1 510 548 5738
  E-mail: *display@usenix.org*

Publications from John Wiley & Sons, Inc.

Member Number: _____

All USENIX members receive a 15% discount on the following Wiley publications

Introduction to Client/Server Systems
Paul Renaud
1-57774-X $34.95 ***member price: $29.71***
# of Copies: _____

Portable UNIX
Douglas Topham
1-57926-2 $14.95 ***member price: $12.71***
# of Copies: _____

The UNIX Command Reference Guide
Kaare Christian
1-85580-4 $32.95 ***member price: $28.01***
# of Copies: _____

C++ For Programmers
Leendert Ammeraal
1-93011-3 $32.95 ***member price: $28.01***
# of Copies: _____

Obfuscated C and Other Mysteries
Don Libes
1-57805-3 $39.95 ***member price: $33.96***
# of Copies: _____

Adventures in UNIX Network Applications Programming
Bill Rieken
1-52858-7 $34.95 ***member price: $29.71***
# of Copies: _____

UNIX, Self Teaching Guide
George Leach
1-57924-6 $19.95 ***member price: $16.96***
# of Copies: _____

UNIX Shell Programming, Second Edition
Lowell Jay Arhur
1-51821-2 $29.95 ***member price: $25.46***
# of Copies: _____

Object Oriented Programming withTurbo C++
Keith Weiskamp
1-52466-2 $24.95 ***member price: $21.21***
# of Copies: _____

Berkeley UNIX
A Simple & Comprehensive Guide
James Wilson
1-61582-X $37.95 ***member price: $32.26***
# of Copies: _____

U.S. orders only

☐ Payment enclosed, plus sales tax
☐ VISA ☐ Mastercard
☐ American Express
Card No. _____

Name _____
Firm _____
Address _____
City/State/Zip _____
Signature _____
(order invalid unless signed)

Please mail or fax orders to the following address:

John Wiley & Sons, Inc.
605 Third Avenue
New York, NY 10158
Attn: Karen Cooper

Phone: (212) 850-6789
FAX: (212) 850-6142

WILEY

Prentice Hall PTR is pleased to recommend the following titles to USENIX members...

____**UNIX System Administration Handbook, Second Edition,** Evi Nemeth/Garth Snyder, 0-13-151051-7
(15105-0) *List:* $48.00 *Members:* $40.80

____**Object-Oriented Modeling and Design,** James Rumbaugh, 0-13-629841-9
(62984-0) *List:* $54.00 *Members:* $45.90

____**Zen and the Art of the Internet, Third Edition,** Brendan Kehoe, 0-13-121492-6
(12149-1) *List:* $23.95 *Members:* $20.36

____**The Magic Garden Explained,** Bernard Goodheart/James Cox, 0-13-098138-9
(09813-7) *List:* $38.00 *Members:* $32.30

____**Internetworking with TCP/IP, Vol. II Design, Implementation, and Internals,** Douglas E. Comer/David L. Stevens, 0-13-472242-6
(47224-1) *List:* $61.33 *Members:* $52.13

____**SCO Performance Tuning Handbook,** Gina Miscovich/David Simons, 0-13-102690-9
(10269-9) *List:* $42.00 *Members:* $35.70

____**Object-Oriented Programming,** Peter Coad/Jill Nicola, 0-13-032616-X
(03261-5) *List:* $48.00 *Members:* $40.80

____**Internetworking with TCP/IP, Vol. III Client Server Programming and Applications for the BSD Socket Version,** Douglas E. Comer and David L. Stevens, 0-13-474222-2
(47422-1) *List:* $53.00 *Members:* $45.05

____**Internetworking with TCP/IP, Vol. III Client Server Programming and Applications for the AT&T TLI Version,** Douglas E. Comer and David L. Stevens, 0-13-474230-3
(47423-9) *List:* $53.00 *Members:* $45.05

____**The Internet Message: Closing the Book with Electronic Mail,** Marshall T. Rose, 0-13-092941-7
(09294-0) *List:* $50.00 *Members:* $42.50

____**The Standard C Library,** PJ Plauger, 0-13-131509-9
(13150-8) *List:* $37.80 *Members:* $32.13

____**All About Administering the NIS+, Second Edition** Rick Ramsey, 0-13-309576-2
(30957-5) *List:* $42.00 *Members:* $35.70

____**The Simple Book: An Introduction to Internet Management** Marshall T. Rose, 0-13-177254-6
(17725-3) *List:* $55.00 *Members:* $46.75

____**Networking Operations on UNIX SVR4,** Mike Padavano, 0-13-613555-2
(61355-4) *List:* $50.00 *Members:* $42.50

____**Solaris Porting Guide,** Sunsoft ISV Engineering 0-13-030396-8
(03039-5) *List:* $52.00 *Members:* $44.20

____**Multiprocessor System Architectures,** Ben Catanzaro 0-13-089137-1
(08913-6) *List:* $44.00 *Members:* $37.40

____**The HP-UX System Administrator's "How To" Book** Marty Poniatowski, 0-13-099821-4
(09982-0) *List:* $32.00 *Members:* $27.20

____**UNIX System V Performance Management,** Edited by Phyllis Eve Bregman and Sally A. Browning 0-13-016429-1
(01642-8) *List:* $29.95 *Members:* $25.45

____**SCO® UNIX® Operating System System Administrator's Guide,** Santa Cruz Operation, 0-13-012568-7
(01256-7) *List:* $39.00 *Members:* $33.15

HERE'S HOW TO ORDER:

CALL
800-880-6818

OR WRITE:
CompuBooks
Route 1, Box 271-D,
Cedar Creek, TX 78612

OR INTERNET:
70007.1333@CompuServe.com
(GO CBK on CompuServe)

WE SHIP ANYWHERE!

FOR MORE INFORMATION, OR QUANTITY ORDERS, PLEASE CALL 201-592-2657

NETWORKS

USENIX members receive a 15% discount on the following MIT Press publications:

GLOBAL NETWORKS
Computers and International Communication
edited by Linda M. Harasim

Global Networks takes up the host of issues raised by the new networking technology that now links individuals, groups, and organizations in different countries and on different continents. The twenty-one contributions focus on the implementation, application, and impact of computer-mediated communication in a global context.

340 pp. $29.95 hardcover HARNH

THE NETWORK NATION
Human Communication via Computer Revised Edition
Starr Roxanne Hiltz and Murray Turoff

"*The Network Nation*... contained a fascinating vision. ...It is a place where thoughts are exchanged easily and democratically and intellect affords one more personal power than a pleasing appearance does. Minorities and women compete on equal terms with white males, and the elderly and handicapped are released from the confines of their infirmities to skim the electronic terrain as swiftly as anyone else." — Teresa Carpenter, *Village Voice*

580 pp. $24.95 paperback HILVP

THE EVOLUTION OF C++
Language Design in the Marketplace of Ideas
edited by Jim Waldo

This collection of articles traces the history of C++, from its infancy in the Santa Fe workshop, to its proliferation today as the most popular object-oriented language for microcomputers. Waldo notes in his postscript that in the process of evolving, the language has lost a clearly articulated, generally accepted design center, with no common agreement about what it should or should not do in the future.

279 pp. $24.95 paperback WALEP

TECHNOLOGY 2001
The Future of Computing and Communications
edited by Derek Leebaert

Researchers, executives, and strategic planners from inside the companies and laboratories that have shaped today's information age forecast the merging technologies that could well define the computing and communications environment that lies ahead.

392 pp. $14.95 paperback LEEEP

THE DIGITAL WORD
Text-Based Computing in the Humanities
edited by George P. Landow and Paul Delany

This book explores the larger realm of the knowledge infrastructure where texts are received, reconstructed, and sent over global networks.

Technical Communication and Information Systems series 384 pp. $39.95 hardcover LANDH

SOCIOMEDIA
Multimedia, Hypermedia, and the Social Construction of Knowledge
edited by Edward Barrett

Sociomedia continues the assessment of hypertext and hypermedia systems begun in *Text, ConText, and HyperText* and *The Society of Text*. It examines the use of integrated multimedia to support social or collaborative research, learning, and instruction in the university, one of the best environments for developing and analyzing the effects of computing technologies on our understanding of complex sets of information.

Technical Communications and Information Series 360 pp. $50.00 hardcover BARRH

CONNECTIONS
New Ways of Working in the Networked Organization
Lee Sproull and Sara Kiesler

"...Sproull and Kiesler raise crucial questions about our technical and particularly our human strategies as a producing society." — Howard Webber, *Sloan Management Review*

228 pp. $21.95 paperback SPRCP

TECHNOBABBLE
John A. Barry

"A serious study of the language of the new technocracy." — William Safire, *The New York Times Magazine*

288 pp. $12.50 paperback BARCP

Please send me these titles _____

☐ Payment enclosed ☐ Purchase Order Attached Charge to my: ☐ Master Card ☐ VISA

Card # _____ exp. _____ Signature _____

$ _____ Total for book(s)
$ 3.00 Postage for North American addresses
$ _____ Canadian customers add 7% GST
$ _____ Total for book(s) & postage

Name _____
Address _____
City _____ State _____ Zip _____
Phone _____ FAX _____

Make checks payable and send order to:

THE MIT PRESS
55 Hayward Street, Cambridge, MA 02142-1399 USA

To order by phone, call (617) 625-8569 or (800) 356-0343. E-mail order # mitpress-orders@mit.edu. The operator will need this code: **UNIX1.**

A UNIQUE OFFER ON THE BEST IN UNIX FOR USENIX MEMBERS

20% DISCOUNT FROM McGRAW-HILL

☐ **THE INTERNET GUIDE FOR NEW USERS**
D. Dern
hardcover, 016510-6, $40.00,
MEMBER PRICE $32.00
paperback, 016511-4, $27.95,
MEMBER PRICE $22.36

☐ **INTERNET FOR EVERYONE**
R. Wiggins
hardcover, 067018-8, $29.95,
MEMBER PRICE $23.96
paperback, 067019-6, $45.00,
MEMBER PRICE $36.00

☐ **THE ESSENTIAL INTERNET INFORMATION GUIDE**
J. Manger
707905-1, paperback, $27.95,
MEMBER PRICE $22.36

☐ **THE INFORMATION BROKERS HANDBOOK, Second Edition**
S. Rugge
911878-x, paperback, $34.95,
MEMBER PRICE $27.96
Available December 1994

☐ **SAA AND UNIX: IBM's Open System Strategy**
M. Killen
034607-0, $40.00,
MEMBER PRICE $32.00

☐ **A STUDENT'S GUIDE TO UNIX**
H. Hahn
025511-3, paperback, $28.00,
MEMBER PRICE $22.40

☐ **UNIX DEVELOPER'S TOOL KIT**
K. Leininger
911836-4, $65.00,
MEMBER PRICE $52.00

☐ **UNIX SECURITY: A Practical Tutorial**
N. Arnold
002560-6, $24.95,
MEMBER PRICE $19.96

☐ **THE UNIX AUDIT: Using UNIX to Audit UNIX**
M. Grottola
025127-4, $32.95,
MEMBER PRICE $26.36

☐ **UNIX: A Database Approach**
S. Das
015745-6, $29.95,
MEMBER PRICE $23.96
Available November 1994

I am a member of USENIX Association. Please send me the books I have indicated at the member special rate. I have added $3.00 postage and handling for the first book ordered, $1.00 for each additional book, plus my local sales tax.

Check or money order is enclosed—payable to McGraw-Hill, Inc.
Charge my ❑ Visa ❑ Mastercard ❑ Discover ❑ Amex

Account #_____

Expiration Date_____

Signature_____

USENIX Membership # _____

Bill & Ship To:

Name_____

Street_____

City, State, Zip_____

Daytime Phone #_____

03US002

Send or Fax Orders to:

McGraw-Hill, Inc.
Attn: Rosa Perez
11 West 19th Street—4th Floor
New York, New York 10011
Fax: 212-337-4092

If I am not completely satisfied, I will return the book(s) within 15 days for a full refund or credit. Satisfaction unconditionally guaranteed. Prices subject to change without notics. We can only accept orders within the continental USA.

FROM O'REILLY & ASSOCIATES, INC.

10% OFF TO USENIX MEMBERS
Mention "Usenix" & your membership number when ordering.

Books that help people get more out of computers

The Whole Internet User's Guide & Catalog

By Ed Krol, 1st Edition September 1992
400 pages, ISBN 1-56592-025-2

A comprehensive—and best-selling—introduction to the Internet, the international network that includes virtually every major computer site in the world.
This book is a comprehensive introduction to what's available and how to find it. In addition to electronic mail, file transfer, remote login, and network news, *The Whole Internet* pays special attention to some new tools for helping you find information. Whether you're a researcher, a student, or just someone who likes electronic mail, this book will help you to explore what's possible.

Learning perl

By Randal L. Schwartz
1st Edition November 1993
274 pages, ISBN 1-56592-042-2

Perl is rapidly becoming the "universal scripting language." Combining capabilities of the UNIX shell, the C programming language, *sed*, *awk*, and various other utilities, it has proved its use for tasks ranging from system administration to text processing and distributed computing. *Learning perl* is a step-by-step, hands-on tutorial designed to get you writing useful perl scripts as quickly as possible. In addition to countless code examples, there are numerous programming exercises, with full answers. For a comprehensive and detailed guide to programming with Perl, read O'Reilly's companion book *Programming perl*.

Understanding Japanese Information Processing

By Ken Lunde
1st Edition September 1993
470 pages, ISBN 1-56592-043-0

Understanding Japanese Information Processing provides detailed information on all aspects of handling Japanese text on computer systems. It covers everything from the origins of modern-day Japanese to the latest information on specific emerging computer encoding standards.

sendmail

By Bryan Costales, with Eric Allman & Neil Rickert
1st Edition November 1993
830 pages, ISBN 1-56592-056-2

Although *sendmail* is used on almost every UNIX system, it's one of the last great uncharted territories—and most difficult utilities to learn—in UNIX system administration. This book provides a complete *sendmail* tutorial, plus extensive reference material. It covers the BSD, UIUC IDA, and V8 versions of *sendmail*. Part One of the book is a tutorial on understanding *sendmail*; Part Two covers practical issues in *sendmail* administration; Part Three is a comprehensive reference section; and Part Four consists of appendices and a bibliography.

ORACLE Performance Tuning

By Peter Corrigan & Mark Gurry
1st Edition September 1993
642 pages, ISBN 1-56592-048-1

The ORACLE relational database management system is the most popular database system in use today. This book shows you the many things you can do to dramatically increase the performance of your existing ORACLE system, whether you are running RDBMS Version 6 or Version 7. You may find that this book can save you the cost of a new machine; at the very least, it will save you a lot of headaches.

Migrating to Fortran 90

By James F. Kerrigan
1st Edition November 1993 (est.)
315 pages (est.), ISBN 1-56592-049-X

This book is a practical guide to Fortran 90 for the current Fortran programmer. It provides a complete overview of the new features that Fortran 90 has brought to the Fortran standard, with examples and suggestions for use. Topics include array sections, modules, allocatable arrays and pointers, file handling, and numeric precision.

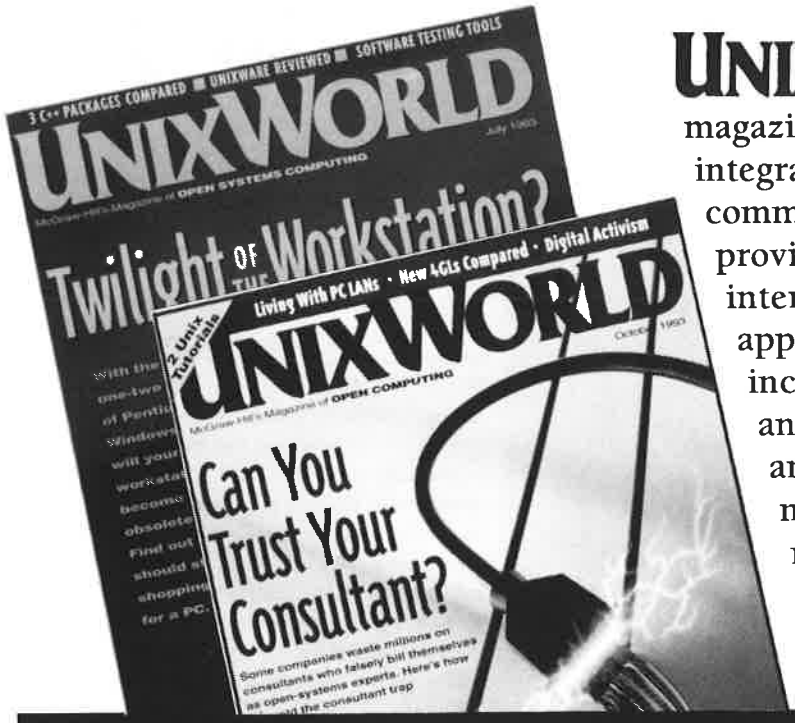
O'Reilly & Associates, Inc.

103A Morris Street, Sebastopol, California 95472 • 800/998-9938 • 707/829-0515 • FAX 707/829-0104 • order@ora.com

Special Offer

USENIX Association Members

THE WORLDWIDE COMPUTING AUTHORITY



UNIXWORLD is a monthly magazine written for people who integrate, resell, manage, and program commercial computer systems to provide solutions based on interoperable networks and applications. Topics for features include the effective use of UNIX and other open systems to downsize and distribute applications. The magazine also provides a useful mix of product reviews, case studies, industry profiles and analysis, and technical tutorials.

THE BENEFITS OF UNIXWORLD:

Programming Tips - helps the reader become educated in UNIX and OPEN SYSTEMS computing. Provides real-world application information the reader can use everyday.

Unbiased Hardware and Software Reviews - helps the reader recommend and justify purchasing decisions.

Clear Editorial - so both beginners and experts can get full benefit from the publication. Product Trends and Technical Advances - to aid professionals in developing company-wide systems strategies.

Profiles on U.S. and International Companies - provides the reader with a truly global perspective of the marketplace and the players. Helps vendors assess and monitor the competition.

Special Rates:

DOMESTIC		CANADIAN		OTHER INTERNATIONAL (Air Delivery)	
1 Yr.	$15.00	1 Yr.	$21.00	1 Yr.	$ 48.00
2 Yrs.	$24.00	2 Yrs.	$36.00	2 Yrs.	$ 90.00
3 Yrs.	$30.00	3 Yrs.	$48.00	3 Yrs.	$130.00

Be wise subscribe today!

To place your order please call:
Kelly Kendall at 1-800-327-6850 or fax your order to 609/426-7055

McGraw-Hill, Inc.• Dept. 418 • Hightstown, NJ 08520

UniForum™'95

The Official Conference and Exposition for Open Computing Solutions

March 14–16, 1995

Dallas Convention Center • Dallas, Texas

REGISTER BY FAX!

Call 617-449-5554, enter Code 70 and have your fax number ready — we'll fax back your registration form within 24 hours!

NEW!

UNIFORUM ON-LINE...for the latest information!

Internet

World Wide Web URL: http://www.uniforum.org

©1994 The Interface Group • 300 First Avenue, Needham, MA 02194-2722 USA

Now managed by The Interface Group, the producer of COMDEX®.

Sponsored by UniForum, The International Association of Open Systems Professionals.

The Second International World-Wide Web (WWW) Conference

MOSAIC AND THE WEB
October 17-20, 1994
Chicago, Illinois
Ramada-Congress Hotel

THE SECOND INTERNATIONAL
WWW CONFERENCE '94
Mosaic and the Web

UniForum, the International Assocation of Open Systems Professionals, is delighted to announce its cooperation with the above conference sponsored by:

- National Center for Supercomputing Applications (NCSA) at the University of Illinois at Urbana-Champaign
- Open Software Foundation (OSF) Research Institute
- National Science Foundation (NSF)
- The European Laboratory for Particle Physics (CERN)

People are excited about Mosaic! — Academicians, government officials and business people will all come together to discuss their WWW and Mosaic activities and tangible results.

This conference is not about a piece of code; it is about a revolution as significant as that engendered by the printing press.

Mosaic and the Web are introducing a major transformation in the way in which knowledge is being expressed and communicated around the world.

Participants will meet other users, talk, share information, and explore the latest and greatest WWW and Mosaic developments.
Sessions will cover information discovery and retrieval, HTML, information security, server administration, and much more.

Emphasis will be given to practical applications, demonstrated techniques, tangible results, existing tools and systems, and the latest research.

The Schedule:
- **Monday** - Tutorials
- **Tuesday & Wednesday** - Conference Sessions, Workshops, Panels, & BOFs (Birds of Feather sessions)
- **Thursday** - Programmers and Developers

To receive a registration packet, send your information to:
Sandy Caldwell (well@osf.org)
OSF Research Institute
One Cambridge Center
Cambridge, MA 02142
617/621-7339; FAX 617/621-8696

Registration and program information may be queried via the web:
<http://riwww.osf.org:8001/ri/announcements/WWW_Conf_F94.html>

UniForum The International Association of Open Systems Professionals

Prime Time Freeware for UNIX

Now available to UniForum members at a discounted rate: Prime Time Freeware for UNIX.

This collection of UNIX-related freeware is published semi-annually by Prime Time Freeware and is now available to UniForum members at a discounted rate. With 5,000 MB of free source code and documentation, each issue contains a book and two ISO-9660 CD-ROMs.

Issue 3-1 contains more than 100 interpreters and compilers and dozens of other software development tools; documentation preparation tools, graphics and GUI tools, math and simulation code, windowing utilities, and much more.

By searching the Internet, PTF brings you the software you need in a convenient format -- and UniForum offers it to you at a price you'll like. To order your copy, call UniForum at 1-800-255-5620 or complete the form and send it back to UniForum with payment today!

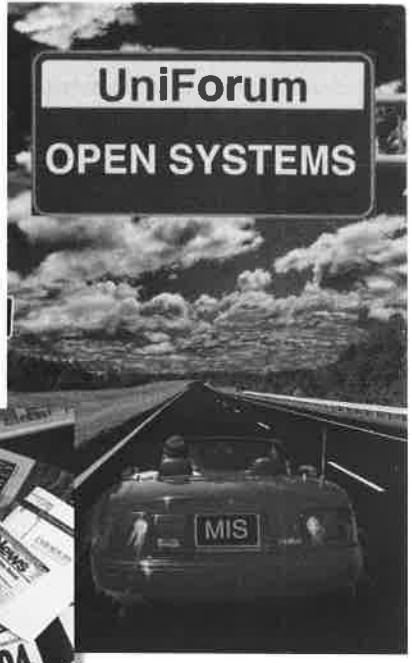
UniForum General Member price	$45
UniForum Associate Member price	$50
UniForum Trial Member price	$60

Send orders to: **UniForum Prime Time Freeware,**
2901 Tasman Drive, Suite 205, Santa Clara, CA 95054-1100
Tel: (408) 986-8840 (800) 255-5620, Fax: (408) 986-1645

Send me ____ copies of Prime Time Freeware for UNIX!
Disk price ($45, $50, $60): _____
Subtotal: _____
Tax (CA residents only): _____
Shipping (see below): _____
Total: _____
Ship to: _____
UniForum Membership No. _____
Company _____
Title _____
Address: _____
City _____
State/Prov. _____ Zip _____
Country _____
Phone _____
Fax _____
Email _____
Method of Payment: ☐ Check payable to UniForum
☐ Money order (in U.S. dollars)
☐ Visa ☐ MasterCard ☐ American Express
Credit Card #: _____ Exp.Date _____
Signature: _____

Payment must be included with all orders and checks must be drawn on a U.S. bank. Credit card orders are also accepted by telephone.
* $5 for 1 to 5 sets. For larger orders contact UniForum for shipping prices.

What is Open Systems and where can you find the information you need to keep up in today's competitive open systems environment?



Find out by joining UniForum: The International Association of Open Systems Professionals.

The Advantages of Membership:

Your annual membership is a resource treasure chest. Among the many valuable benefits you'll receive are:

- *UniForum Monthly*: The Magazine for Open Systems Professionals
- *UniNews* - the authoritative voice of UniForum - now on-line
- *The Open Systems Products Directory* - comprehensive, accurate, indispensable
- Reduced fees at all UniForum conferences and seminars
- Technical and Standards publications: timely, reliable and useful
- Discounts on many outstanding industry publications
- Discounts on leading commercial hardware and software products
- Discounts on courses offered by leading training organizations
- Discounts on exclusive new shareware selections
- Services such as discounts on car rentals, travel and mail list rentals
- Eligibility to serve on UniForum committees and the Board of Directors

UNIFORUM MEMBERSHIP APPLICATION FORM *(please print)*

Name _____ MR/MS

Title _____

Company _____

Address _____

Mail Stop _____

City _____ State _____ Zip _____

Country _____ Telephone (_____)

E-mail (where we will send UniNews) _____ Fax (_____)

General Membership** $125

Includes *UniForum Monthly* magazine, *UniNews* electronic newsletter, *Open Systems Products Directory,* technical publications, recruitment advertising service, and discounts on: UniForum conference registration and shareware CD-ROMs, purchases of software and hardware, educational seminars and special classes, additional UniForum publications and other industry publications, computer books, CD-ROMs and training services.

** Overseas postage:
air $100 or surface $60 (no additional postage necessary for Canada, Mexico or Puerto Rico) $_____

TOTAL AMOUNT ENCLOSED $_____

Method of Payment:

Select one: ❑ Check payable to UniForum * ❑ Money Order In US Dollars
❑ MasterCard ❑ Visa ❑ American Express

Credit card number _____ Expiration date _____

Signature _____

*Payments must be included in U.S. dollars. All checks must be drawn on a U.S. bank. Credit card orders can be accepted by telephone.
**Overseas surface delivery takes up to six weeks.

Send application and payment to: UniForum, 2901 Tasman Drive, Suite 205, Santa Clara, CA 95054-1100
Tel: (408) 986-8840 (800) 255-5620 Fax: (408) 986-1645

Prices subject to change without notice. Contact UniForum for discounts and shipping and handling charges on bulk orders. UniForum is a registered trademark of UniForum. UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Limited.

UniForum
The International Association of Open Systems Professionals

There's Never Been a Better Time for an Outstanding Value

As the unstoppable move to open systems gathers even more momentum you need a strong and independent association that can help you achieve your professional goals. An association that can magnify your voice, which can educate, inform and inspire. You need and deserve an association that provides value for your money. In short you need UniForum. With a value this good, the real question might be, "Can you afford to be without it?" Join today!

The Association will support local user groups by doing a mailing to assist in the formation of a new group and publishing information on local groups in *;login:*. At least one member of the group must be a current member of the Association. Send additions and corrections to: <*login@usenix.org*>.

California

Fresno:

The Central California UNIX Users Group consists of a uucp-based electronic mailing list to which members may post questions or information. For connection information:

- Educational and governmental institutions:
  Brent Auernheimer
  (209) 278-2573,
  <*brent@CSUFresno.edu*> or
  <*csufres!brent*>
- Commercial institutions or individuals:
- Gordon Crumal
  (209) 251-2648
  <*csufres!gordon*>

Orange County

Meets the 2nd Monday of each month

- UNIX Users Association of Southern California
  Paul Muldoon
  (714) 556-1220 ext. 137
  New Horizons Computer Learning Center
  1231 E. Dyer Rd., Suite 140
  Santa Ana, CA 92705

Colorado

Boulder

Meets monthly at different sites. For meeting schedule, send email to <*fruug-info@fruug.org*>.

- Front Range UNIX Users Group
  Lone Eagle Systems Inc.
  636 Arapahoe #10
  Boulder, CO 80302
  Steve Gaede
  (303) 444-9114
  <*gaede@fruug.org*>

Washington, D.C.

Meets 1st Tuesday of each month.

- Washington Area UNIX Users Group
  9811 Mallard Drive
  Laurel, MD 20708
  Alan Fedder
  (301) 953-3626

Florida

Coral Springs:

- S. Shaw McQuinn
  (305) 344-8686
  8557 W. Sample Road
  Coral Springs, FL 33065

Melbourne:

Meets the 3rd Monday of every month.

- Space Coast UNIX User's Group
  Steve Lindsey
  (407) 242-4766
  <*lindsey@vnet.ibm.com*>

Orlando:

Meets the 3rd Thursday of each month.

- Central Florida UNIX Users Group Mikel Manitius
  (407) 444-8448
  <*mikel@aaa.com*>

Western:

Meets 1st Thursday of each month.

- Florida West Coast UNIX Users Group
  Richard Martino (813) 536-1776
  Tony Becker (813) 799-1836
  <*mcrsys!tony*>
  Ed Gallizzi, Ph.D. (813) 864-8272
  <*e.gallizzi@compmail.com*>
  Jay Ts (813) 979-9169
  <*uunet!pdn!tscs!metran!jan*>
  Dave Lewis (407)242-4372
  <*dhl@ccd.harris.com*>

Georgia

Atlanta:

Meets on the 1st Monday of each month in White Hall, Emory University.

- Atlanta UNIX Users Group
  P.O. Box 12241
  Atlanta, GA 30355-2241
  Mark Landry (404) 365-8108

Kansas or Missouri

Meets on 2nd Tuesday of each month.

- Kansas City UNIX Users Group (KCUUG)
  P.O. Box 412622
  Kansas City, MO 64141
  (816) 891-1093
  <*richj@northcs.cps.com*>

Michigan

Detroit/Ann Arbor

Meets on the 2nd Thursday of each month in Ann Arbor.

- Southeastern Michigan Sun Local Users Group and Nameless UNIX Users Group
  Steve Simmons office:
  (313)769-4086
  home: (313) 426-8981
  <*scs@lokkur.dexter.mi.us*>

Minnesota

Minneapolis/St. Paul:

Meets the 1st Wednesday of each month.

- UNIX Users of Minnesota
  17130 Jordan Court
  Lakeville, MN 55044
  Robert A. Monio
  (612) 220-2427
  <*pnessutt@dmshq.mn.org*>

Missouri

St. Louis:

- St. Louis UNIX Users Group P.O. Box 2182 St. Louis, MO 63158
  Terry Linhardt
  (314) 772-4762
  <*uunet!jgaltstl!terry*>

Nebraska

Omaha: Meets monthly.

- /usr/group/nebraska
  P.O. Box 31012
  Omaha, NE 68132
  Phillip Allendorfer
  (402) 423-1400

New England

Northern:
Meets monthly at different sites.

- Peter Schmitt (603) 646-2085
  Kiewit Computation Center
  Dartmouth College
  Hanover, NH 03755
  <peter.schmitt@dartmouth.edu>

New Jersey

Princeton:
Meets monthly.

- Princeton UNIX Users Group
  Mercer County Community College
  1200 Old Trenton Road
  Trenton, NJ 08690
  Peter J. Holsberg (609) 586-4800
  <mccc!pjh>

New Mexico

Albuquerque:
ASIGUNIX meets every 3rd Wednesday of each month.

- Phil Hortz 505/275-0466.

New York

New York City:
Meets every other month in Manhattan.

- Unigroup of New York City
  G.P.O.
  Box 1931
  New York, NY 10116
  <unigroup@murphy.com>
  Bob Young (212) 490-8470

Oklahoma

Tulsa:
Meets 2nd Wednesday of each month.

- Yulsa UNIX Users Group, $USR
  Stan Mason (918) 560-5329
  <tulsix!smason@drd.com>
  Mark Lawrence (918) 743-3013
  <mark@drd.com>

Texas

Austin:
Meets 3rd Thursday of each month.

- Capital Area Central Texas UNIX Society (CACTUS)
  P.O. Box 9786
  Austin, TX 78766-9786
  Tom Painter (512) 258-7321
  <president@cactus.org>

Dallas/Fort Worth:
Meets the 1st Thursday of each month.

- Dallas/Fort Worth UNIX Users Group
  P.O. Box 867405
  Plano, TX 75086
  Evan Brown (214) 519-3577
  <evbrown@dsccc.com>

Houston:
Meets 3rd Tuesday of each month.

- Houston UNIX Users Group
  (Hounix) answering machine
  (713) 684-6590
  Bob Marcum, President
  (713) 626-4100
  Chuck Bentley, Vice-president
  (713) 789-8928
  <chuckb@hounix.uucp>

Washington

Seattle:
Meets monthly.

- Seattle UNIX Group Membership Info.
  Bill Campbell (206) 947-5591
  6641 East Mercer
  Mercer Island, WA 98040-0820
  <bill@celestial.com>

Canada

Manitoba:
Meets 2nd Tuesday of each month.

- Manitoba UNIX User Group
  (MUUG) P.O. Box 130
  St. Boniface Winnipeg,
  MB R2H 3B4
  Bary Finch, President
  (204) 934-2723
  <info@muug.mb.ca>

Ottawa:

- The Ottawa Carleton UNIX Users Group
  D.J. Blackwood
  (613)957-9305
  <dave@revcan.rct.ca>

Toronto:

- 143 Baronwood Court
  Brampton, Ontario
  Canada L6V 3H8
  Evan Leibovitch
  (416) 452-0504
  <evan@telly.on.ca>

System Administration Groups

Back Bay LISA (BBLISA)

New England forum covering all aspects of system and network administration, for large and small installations. Meets monthly, at MIT in Cambridge, MA.
For information, contact:

- J. R. Oldroyd (617)227-563
  <jr@opal.com>
- Mailing list subscription:
  <requests:bblisaquest@cs.umb.edu>
- Mailing list postings:
  <bblisa@cs.umb.edu>
- For current calendar of events:
  finger *<bblisa@cs.umb.edu>*

Bay LISA

Meets 3rd Thursday of each month in Mountain View, CA For more information, please contact:
<baylisa-info@baylisa.org> or

- Bryan McDonald,
  BayLISA President
  <bigmac@baylisa.org>
  P.O. Box 64369
  Sunnyvale CA, 94088-4369
  (415) 859-3246

$GROUPNAME (New Jersey)

$GROUPNAME is an organization in New Jersey formed to facilitate information exchange pertaining to the field of UNIX system administration. For more information, send email to:
Majordomo@Warren. MENTORG.COM or
Tom Limoncelli
<tom_limoncelli@warren.mentorg.com>

New York Systems Administrators (NYSA)

Meets 2nd Monday of each month.

- *<nysa-request@esm.com>*
  914/472-3635

North Carolina System Administrators Group

The North Carolina System Administrators Group meets on the 2nd Monday each month around the Research Triangle Park area.

- Amy Kreiling (919) 962-1843
  <kreiling@cs.unc.edu>
- William E. Howell (919) 962-1717
  <howell@cs.unc.edu>

ACM:
Association for Computing Machinery

AFUU:
Association of French UNIX Users

ASPLOS:
Architectural Support for Programming Languages and Operating Systems

AUUG:
Australian UNIX Users Group

COOTS:
Conference on Object-Oriented Technology

DECUS:
Digital Equipment Computer Users Society

EurOpen:
European Forum for Open Systems

FedUNIX:
Council of Advanced Computing Systems Technologists in Government

GURU:
Roumanian UNIX User Group

GUUG:
German UNIX Users Groups

IEEE: Institute of Electrical and Electronics Engineers

IETF:
Internet Engineering Task Force

INET:
Internet Society

Interex:
International Association of Hewlett-Packard Computer Users

JUS:
Japan UNIX Society

LISA:
USENIX Systems Administration Conference

NOSSDAV:
Network and OS Support for Digital Audio and Video

OOPSLA:
Object-oriented Programming Systems, Languages and Applications

OSDI:
Symposium on Operating Systems Design & Implementation

SAGE:
System Administrators Guild

SANS:
System Administration, Networking & Security

SUG:
Sun User Group

SUUG:
Society of Russia UNIX Users Group

UKUUG:
United Kingdom UNIX Systems Users Group

UniForum:
International Association of UNIX and Open Systems Professionals

CALENDAR OF EVENTS

This is a combined calendar of planned conferences, symposia, and standards meetings related to the UNIX operating system. If you have a UNIX-related event that you wish to publicize, please contact <*login@usenix.org*>. Please provide your information in the same format as above.

* = events sponsored by the USENIX Association.

1994

October
4 - 7 ASPLOS VI, San Jose, CA
4 - 6 UNIX Expo, New York City
12-13 AFUU, Grenoble, France
17-21 IEEE 1003, Seattle, WA
23-27 ACM OOPSLA, Portland, OR
26-28* Very High Level Languages, Santa Fe, NM

November
3- 4 GURU, Bucharest, Romania
14-18* OSDI, Monterey, CA
14-18 SUG Technical Workshop, Austin, TX

December
8 - 9* IEEE Mobile Computing Systems & Applications, Santa Cruz, CA
10-15 DECUS, Anaheim. CA

1995

January
9-13 IEEE 1003
16-20* USENIX, New Orleans, LA

March
13-17 UniForum, Dallas, TX

April
10-11* 2nd Mobile and Location–Independent Computing, Ann Arbor, MI
10-14 IEEE 1003
24-29 * SANS IV, Washington, DC

May
6-11 DECUS, Washington, DC

June
5 - 7 * UNIX Security, Salt Lake City, UT
26 - 29* COOTS, Monterey, CA

July
10-14 IEEE 1003

August
6-11 Siggraph, Los Angeles, CA
13-17 Interex 95, Toronto, Canada

September
18-22* LISA IX, Monterey, CA
19-21 UNIX Expo, New York City

October
9-13 IEEE 1003

November
2- 8 DECUS, San Francisco, CA

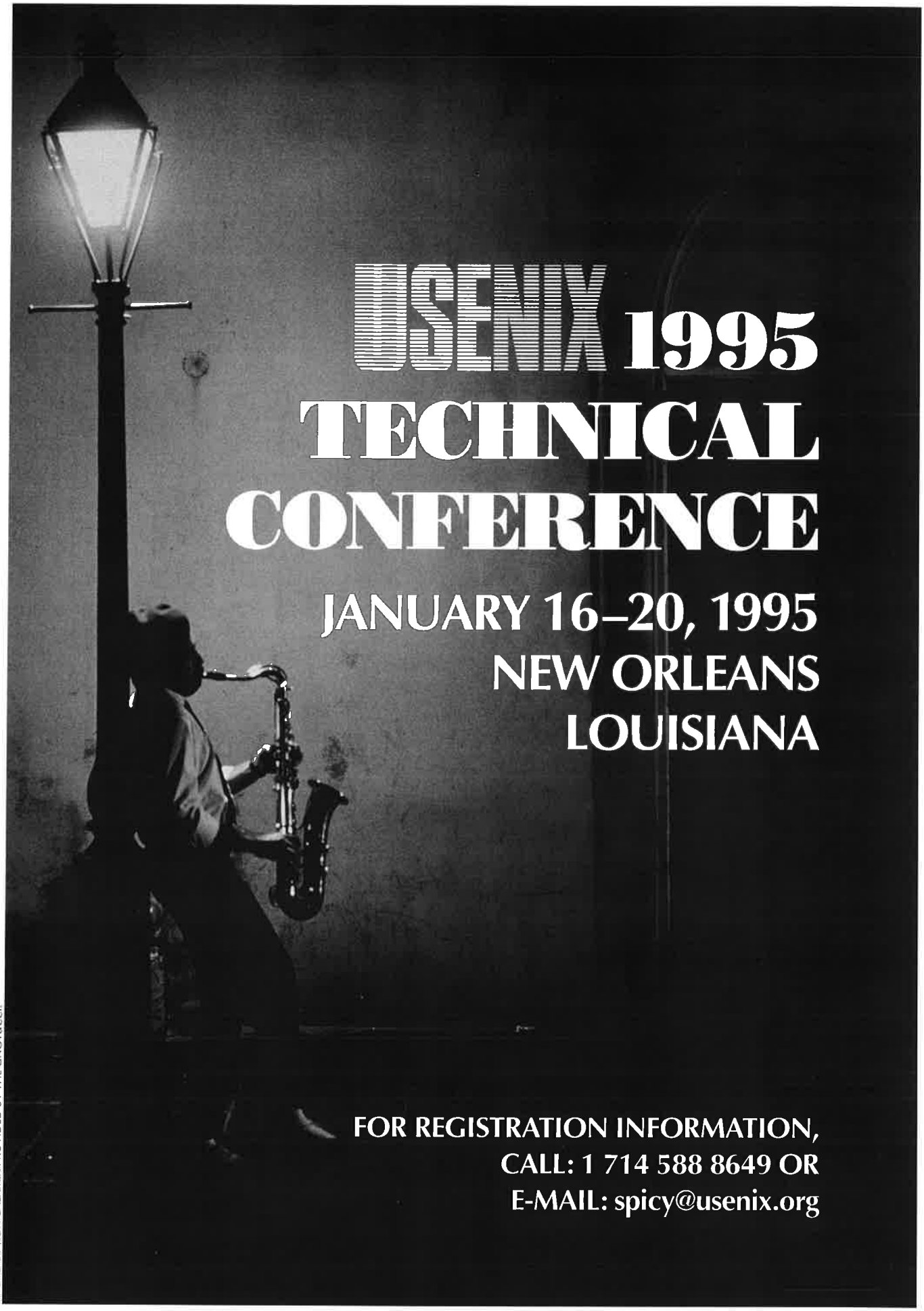
1996

January
22-26* USENIX, San Diego, CA

February
14-16 UniForum, San Francisco, CA

August
4 - 8 Interex 96, San Diego, CA

November
16-22 DECUS, Anaheim, CA



USENIX 1995 TECHNICAL CONFERENCE

JANUARY 16–20, 1995
NEW ORLEANS
LOUISIANA

FOR REGISTRATION INFORMATION,
CALL: 1 714 588 8649 OR
E-MAIL: spicy@usenix.org

PHOTO BY RON CALAMIA PROVIDED BY THE GNOT&CCL.

SECOND CLASS POSTAGE
PAID
AT BERKELEY, CALIFORNIA
AND ADDITIONAL OFFICES

USENIX Association
2560 Ninth Street
Suite 215
Berkeley, CA 94710

POSTMASTER:
SEND ADDRESS CHANGES
TO ***;login:***
USENIX ASSOCIATION
2560 NINTH STREET
SUITE 215
BERKELEY, CA 94710